

# **QUADRATURE ENCODER INPUT DUAL PROPORTIONAL VALVE HIGH TEMPERATURE CONTROLLER**

With SAE J1939

## **USER MANUAL**

**P/N: AX023240**     *J1939 250kbits/s Baud Rate*

**P/N: AX023240-01** *J1939 500kbits/s Baud Rate*

**P/N: AX023240-02** *J1939 1Mbits/s Baud Rate*

## ACCRONYMS

ACK	Positive Acknowledgement (from SAE J1939 standard)
EA	Electronic Assistant, p/n AX070502 (A Service Tool for Axiomatic ECUs)
ECU	Electronic Control Unit (from SAE J1939 standard)
NAK	Negative Acknowledgement (from SAE J1939 standard)
PDU1	A format for messages that are to be sent to a destination address, either specific or global (from SAE J1939 standard)
PDU2	A format used to send information that has been labeled using the Group Extension technique, and does not contain a destination address.
PGN	Parameter Group Number (from SAE J1939 standard)
PropA	Message that uses the Proprietary A PGN for peer-to-peer communication
PropB	Message that uses a Proprietary B PGN for broadcast communication
SPN	Suspect Parameter Number (from SAE J1939 standard)

# TABLE OF CONTENTS

<b>1. OVERVIEW OF CONTROLLER</b> .....	<b>6</b>
<b>1.1. DESCRIPTION OF HIGH TEMPERATURE QUADRATURE ENCODER INPUT DUAL OUTPUT VALVE CONTROLLER</b> .....	<b>6</b>
<b>1.2. QUADRATURE ENCODER INPUT FUNCTION BLOCK</b> .....	<b>7</b>
1.2.1. Input Sensor Types .....	7
1.2.2. Pullup / Pulldown Resistor Options .....	8
1.2.3. Minimum and Maximum Error/Ranges .....	8
1.2.4. Input Software Filter Types .....	9
<b>1.3. INTERNAL FUNCTION BLOCK CONTROL SOURCES</b> .....	<b>9</b>
<b>1.4. OUTPUT DRIVE FUNCTION BLOCKS</b> .....	<b>11</b>
<b>1.5. LOOKUP TABLE FUNCTION BLOCK</b> .....	<b>15</b>
1.5.1. Auto Update on Setpoint Changes .....	15
1.5.2. X-Axis, Input Data Response .....	15
1.5.3. Y-Axis, Lookup Table Output .....	16
1.5.4. Default Configuration, Data Response .....	16
1.5.5. Point To Point Response .....	16
1.5.6. X-Axis, Time Response .....	18
<b>1.6. MATH FUNCTION BLOCK</b> .....	<b>18</b>
<b>1.7. PROGRAMMABLE LOGIC FUNCTION BLOCK</b> .....	<b>20</b>
1.7.1. Conditions Evaluation .....	21
1.7.2. Table Selection .....	22
1.7.3. Logic Block Output .....	23
<b>1.8. PID CONTROL FUNCTION BLOCK</b> .....	<b>24</b>
<b>1.9. SET / RESET FUNCTION BLOCK</b> .....	<b>25</b>
<b>1.10. DIAGNOSTIC TROUBLE CODE (DTC) REACT</b> .....	<b>26</b>
<b>1.11. CAN TRANSMIT MESSAGE FUNCTION BLOCK</b> .....	<b>27</b>
1.11.1. CAN Transmit Message Setpoints .....	27
1.11.2. CAN Transmit Signal Setpoints .....	27
<b>1.12. CAN RECEIVE FUNCTION BLOCK</b> .....	<b>27</b>
<b>1.13. DIAGNOSTIC FUNCTION BLOCKS</b> .....	<b>28</b>
<b>2. INSTALLATION INSTRUCTIONS</b> .....	<b>32</b>
<b>2.1. DIMENSIONS AND PINOUT</b> .....	<b>32</b>
<b>2.2. MOUNTING INSTRUCTIONS</b> .....	<b>33</b>
<b>3.1. INTRODUCTION TO SUPPORTED MESSAGES</b> .....	<b>34</b>
<b>3.2. NAME, ADDRESS AND SOFTWARE ID</b> .....	<b>35</b>
<b>4.1. J1939 NETWORK SETPOINTS</b> .....	<b>37</b>
<b>4.2. MISCELLANEOUS SETPOINTS</b> .....	<b>37</b>
<b>4.3. QUADRATURE ENCODER PARAMETER SETPOINTS</b> .....	<b>38</b>
<b>4.4. QUADRATURE ENCODER / DIGITAL INPUT SETPOINTS</b> .....	<b>38</b>
<b>4.5. CONSTANT DATA LIST SETPOINTS</b> .....	<b>39</b>
<b>4.6. LOOKUP TABLE SETPOINTS</b> .....	<b>39</b>
<b>4.7. PROGRAMMABLE LOGIC SETPOINTS</b> .....	<b>40</b>
<b>4.8. MATH FUNCTION SETPOINTS</b> .....	<b>42</b>
<b>4.9. SET-RESET FUNCTION BLOCK SETPOINTS</b> .....	<b>42</b>
<b>4.10. CAN TRANSMIT SETPOINTS</b> .....	<b>43</b>
<b>4.11. CAN RECEIVE SETPOINTS</b> .....	<b>43</b>
<b>5. REFLASHING OVER CAN WITH EA® BOOTLOADER</b> .....	<b>44</b>
<b>6. TECHNICAL SPECIFICATIONS</b> .....	<b>49</b>
<b>6.1. POWER SUPPLY, CAN AND REFERENCE VOLTAGE</b> .....	<b>49</b>
<b>6.2. INPUTS</b> .....	<b>49</b>
<b>6.3. OUTPUTS</b> .....	<b>50</b>
<b>6.4. GENERAL SPECIFICATIONS</b> .....	<b>50</b>
<b>7. VERSION HISTORY</b> .....	<b>52</b>

## LIST OF FIGURES

Figure 1: Hardware Functional Block Diagram .....	6
Figure 2: Quadrature Encoder Signals (QA&QB) and resulting Direction and Step count .....	7
Figure 3: Analog source to Digital input .....	11
Figure 4: Hotshot Digital Profile .....	12
Figure 5: Lookup Table with "Ramp To" Data Response .....	17
Figure 6: Lookup Table with "Jump To" Data Response .....	17
Figure 7: Programmable Logic Function Block .....	20
Figure 8: Programmable Logic Flowchart .....	21
Figure 9: Housing Dimensions .....	32
Figure 10: Screen capture of Default Miscellaneous Setpoints:.....	37
Figure 11: Screen capture of Default Quadrature Encoder Parameter Setpoints .....	38
Figure 12: Screen capture of Default Digital Input 1 Setpoints .....	38
Figure 13: Screen capture of Default Constant Data List Setpoints .....	39
Figure 14: Screen capture of Default Lookup Table 1 Setpoints .....	40
Figure 15: Screen capture of Default Programmable Logic Setpoints .....	41
Figure 16: Screen capture of Default Math Function 1 Setpoints .....	42
Figure 17: Screen capture of Default Set-Reset Block 1 Setpoints .....	42
Figure 18: Screen capture of Default CAN Transmit 1 Setpoints .....	43
Figure 19: Screen capture of Default CAN Receive 1 Setpoints .....	43

## LIST OF TABLES

Table 1: Digital Input Sensor Type Options .....	8
Table 2: Pullup/Pulldown Resistor Options .....	8
Table 3: Input Filtering Types .....	9
Table 4: Control Source Options .....	10
Table 5: Control Source Number Options .....	10
Table 6: Output Type Options.....	11
Table 7: Digital Output Responses .....	12
Table 8: Enable Response Options .....	14
Table 9: Override Response Options.....	14
Table 10: Fault Response Options .....	14
Table 11: Math Function Operators .....	19
Table 12: Condition Operator Options .....	22
Table 13: Condition Evaluation Results .....	22
Table 14: Conditions' Logical Operator Options .....	22
Table 15: Conditions Evaluation Based on Selected Logical Operator.....	22
Table 16: Programmable Logic Block Default Lookup Tables.....	23
Table 17: PID Response Options.....	24
Table 18: Set-Reset Function Block Operation.....	25
Table 19: Fault Detect Thresholds.....	29
Table 20: Lamp Set by Event in DM1 Options.....	30
Table 21: FMI for Event Used in DTC Options .....	31
Table 22: Low Fault FMIs and corresponding High Fault FMIs .....	31
Table 23: Connector Pinout .....	32

# 1. OVERVIEW OF CONTROLLER

## 1.1. Description of High Temperature Quadrature Encoder Input Dual Output Valve Controller

This User Manual describes the architecture and functionality of the High Temperature Quadrature Encoder Input Dual Valve controller.

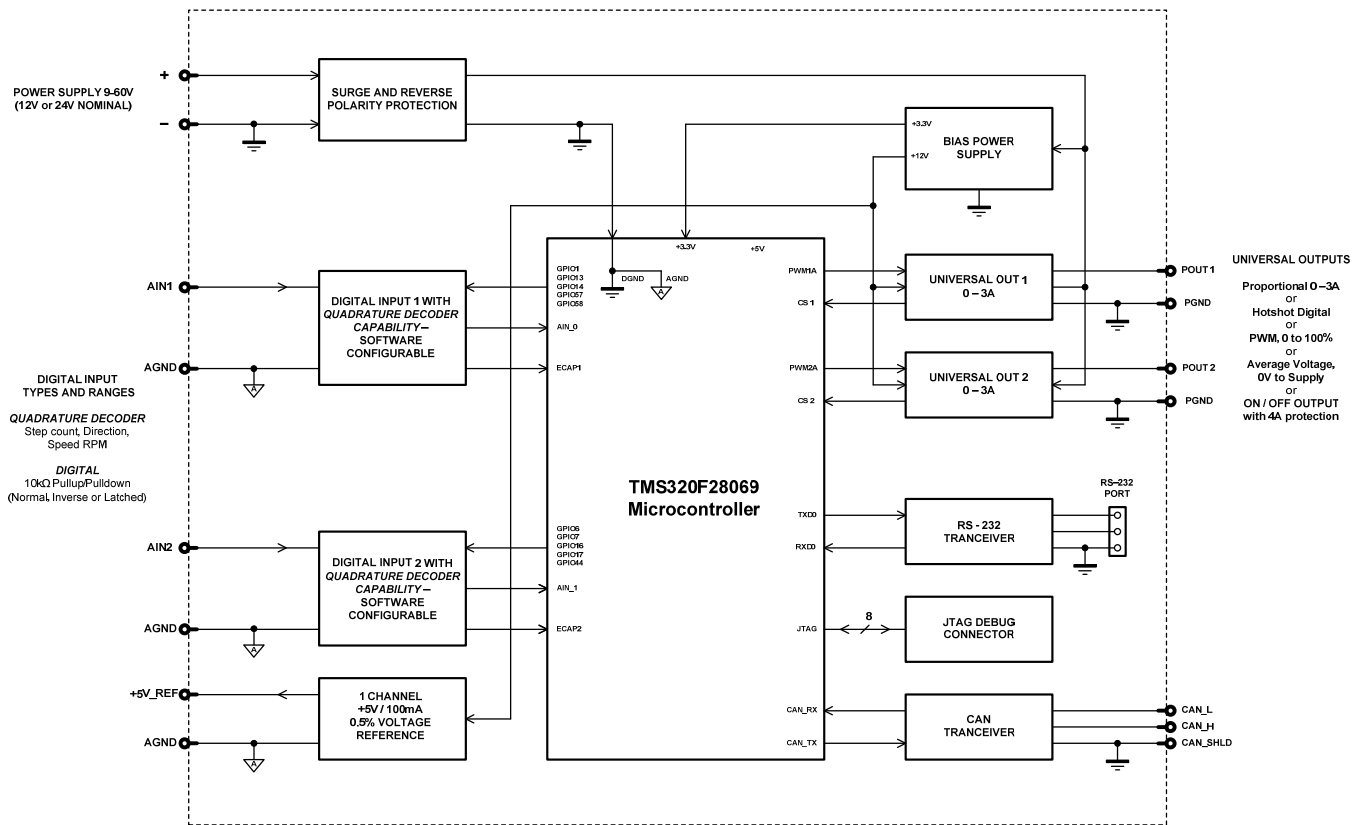


Figure 1: Hardware Functional Block Diagram

The High Temperature Quadrature Encoder Input 2 Output controller is a highly configurable controller with versatile control of 2 digital inputs targeted for quadrature encoder interfacing and 2 proportional valve outputs. Its flexible hardware design allows the controller to have a wide range of output types. The sophisticated control algorithms/logical function blocks allow the user to configure the controller for a wide range of applications without the need for custom firmware.

The 2 digital inputs can be configured to read signals generated by a quadrature Encoder unit. The inputs are described in more detail in section 1.2.

The 2 outputs can be configured to different types: *Proportional Current, Voltage, PWM, Hotshot Digital Current and Digital (ON/OFF)*. Each output consists of a high side half-bridge driver able to source up to 2.5Amps with hardware shutdown at 4Amps. The outputs are described in more detail in section 1.4.

The controller also offers a variety of logical/mathematical functions blocks that can be used to perform application-specific logic or calculations. These functional blocks are explained in more detailed in section 1.5 through section 1.9.

Both outputs and the various logical function blocks on the unit are inherently independent from one another but can be configured to interact in a large number of ways with each other.

The different blocks described above are configured through a Windows-based Axiomatic tool, Electronic Assistant® (EA) via CAN using a USB-CAN converter device (PN: AX070501). Configurable parameters (setpoints) changes made to the High Temperature Quadrature Encoder Input 2 Output controller can be saved in a file (using EA) in order to easily replicate the configuration in any other High Temperature Quadrature Encoder Input 2 Output controller. Electronic Assistant setpoints are described in more detail in sections 3 and 4.

## 1.2. Quadrature Encoder Input Function Block

The controller consists of two digital inputs. The two inputs can be configured to measure digital signals generated by a quadrature Encoder unit.

The High Temperature Quadrature Encoder Input 2 Output controller has configuration options for specifying various Quadrature Encoder unit parameters, such as step count scaler, direction/polarity of rotation, static step count offset and number of quadrature Encoder pulses per revolution (for speed measurements).

The controller has two inputs for detecting the pulses generated by a quadrature Encoder unit. The three measurements (step count, direction and speed) are determined using these two input signals.

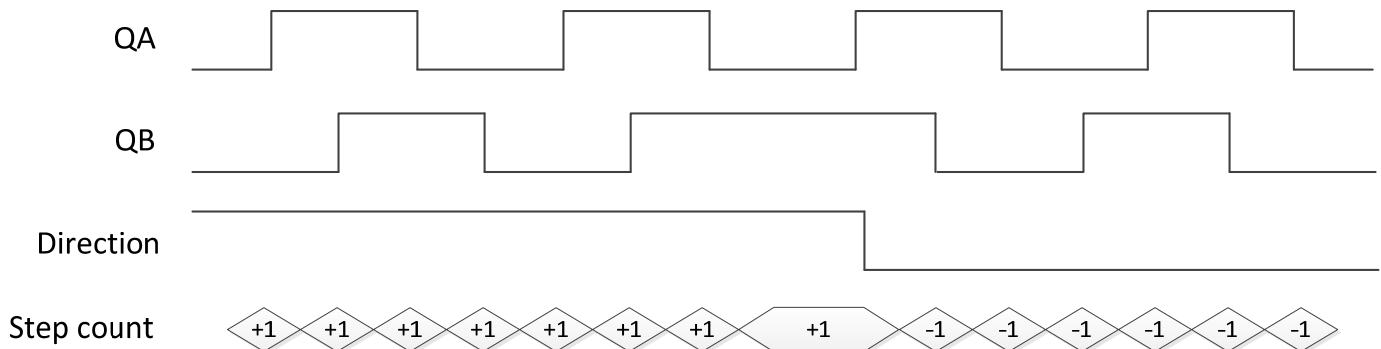


Figure 2: Quadrature Encoder Signals (QA&QB) and resulting Direction and Step count

### 1.2.1. Input Sensor Types

Table 1 lists the supported input types by the controller. The **Input Sensor Type** parameter provides a dropdown list with the input types described in Table 1. Changing the **Input Sensor Type** affects other setpoints within the same setpoint group such as Minimum/Maximum Error/Range by refreshing them to new input type and thus should be changed first.

Table 1: Digital Input Sensor Type Options

0	Disabled
60	Digital (Normal)
61	Digital (Inverse)
62	Digital (Latched)
80	Quadrature Encoder Step Count
81	Quadrature Encoder Direction
82	Quadrature Encoder Speed

Digital **Input Sensor Types** offers three modes: Normal, Inverse, and Latched. The measurements taken with digital input types are 1 (ON) or 0 (OFF).

When a quadrature Encoder unit is read using the High Temperature Quadrature Encoder Input 2 Output controller, both digital inputs should be connected to read the signals generated by the quadrature Encoder unit. Both inputs would need to be configured as one of the Quadrature Encoder input types, Step Count (80), Direction (81) or Speed (82, measured as RPM).

### 1.2.2. Pullup / Pulldown Resistor Options

In all **Input Sensor Types**: Digital and Quadrature Encoder types, the user has the option of three (3) different pull up/pull down options as listed in Table 2.

Table 2: Pullup/Pulldown Resistor Options

0	Pullup/Pulldown Off
1	1kΩ Pullup
2	10kΩ Pulldown

In order to create ‘Active High’/‘Active Low’ configurations – a proper combination of Digital Input modes: *Normal*, *Inverse*, *Latched* and **Pullup/Pulldown Resistor: 1kΩ Pullup, 10kΩ Pulldown** needs selected. For example when using a ‘floating’ input, in order to create an ‘Active Low’ configuration use **Pullup/Pulldown Resistor: 1kΩ Pullup** and **Input Sensor Type: Digital (Inverse)**. The pullup resistor will create a 1 (ON) value when the input is floating. Since it is placed in *Digital (Inverse)*, this value will be considered as OFF. Once the input is grounded this will create a 0 (OFF) but since the input is *Digital (Inverse)* this be considered as ON.

### 1.2.3. Minimum and Maximum Error/Ranges

The **Minimum Range** and **Maximum Range** setpoints are used to create the overall useful range of the inputs. For example if **Minimum Range** is set to 100 RPM and **Maximum Range** is set to 1900 RPM, the overall useful range (0-100%) is between 100 to 1900 RPM. Anything below the **Minimum Range** will saturate at **Minimum Range**. Similarly, anything above the **Maximum Range** will saturate at **Maximum Range**. The other two setpoints, **Minimum Error** and **Maximum Error** have to be lower than the **Minimum Range** and the **Maximum Range** setpoints, respectively. Anything below the **Minimum Error** or above the **Maximum Error** will create a fault. If that input is commanding an output, the output will shut off. If requiring the output to remain active, the **Minimum Error/Minimum Range** and **Maximum Range/Maximum Error** should be

set to the same values. Refer to section 1.13 for diagnostics that can be associated with Input Function Block

#### 1.2.4. Input Software Filter Types

All input types can be filtered using **Filter Type** and **Filter Constant** setpoints. There are three (3) filter types available as listed in Table 3.

*Table 3: Input Filtering Types*

0	<i>No Filtering</i>
1	<i>Moving Average</i>
2	<i>Repeating Average</i>

The first filter option *No Filtering*, provides no filtering to the measured data. Thus the measured data will be directly used to the any function block which uses this data.

The second option, *Moving Average*, applies the 'Equation 1' below to measured input data, where  $Value_N$  represents the current input measured data, while  $Value_{N-1}$  represents the previous filtered data. The Filter Constant is the **Filter Constant** setpoint.

Equation 1 - Moving Average Filter Function:

$$Value_N = Value_{N-1} + \frac{(Input - Value_{N-1})}{Filter\ Constant}$$

The third option, *Repeating Average*, applies the 'Equation 2' below to measured input data, where N is the value of **Filter Constant** setpoint. The filtered input, Value, is the average of all input measurements taken in N (**Filter Constant**) number of reads. When the average is taken, the filtered input will remain until the next average is ready.

Equation 2 - Repeating Average Transfer Function:

$$Value = \frac{\sum_0^N Input_N}{N}$$

### 1.3. Internal Function Block Control Sources

The High Temperature Quadrature Encoder Input 2 Output controller allows for internal function block sources to be selected from the other function blocks supported by the controller. As a result, any output from one function block can be selected as the control source for another. Keep in mind that not all options make sense in all cases, but the complete list of control sources is shown in Table 4.

Table 4: Control Source Options

Value	Meaning
0	Control Not Used
1	Received CAN Message
2	Quadrature Encoder Input Measured
3	Output Target Value
4	Output Current Feedback
5	Lookup Table
6	Math Function Block
7	Programmable Logic Block
8	PID Function Block
9	Control Constant Data
10	Set/Reset Block
11	Diagnostic Trouble Code
12	Power Supply Measured
13	Processor Temperature Measured

In addition to a source, each control also has a number which corresponds to the sub-index of the function block in question. Table 5 outlines the ranges supported for the number objects, depending on the source that had been selected.

Table 5: Control Source Number Options

Control Source	Control Source Number Range
Control Not Used	[0]
Received CAN Message	[1...5]
Quadrature Encoder Input Measured	[1...2]
Output Target Value	[1...2]
Output Current Feedback	[1...2]
Lookup Table	[1...3]
Math Function Block	[1...2]
Programmable Logic Block	[1...1]
PID Function Block	[1...1]
Control Constant Data	[1...10]
Set/Reset Block	[1...2]
Diagnostic Trouble Code	[1...3]
Power Supply Measured	[1...1]
Processor Temperature Measured	[1...1]

If a non-digital signal is selected to drive a digital input, the signal is interpreted to be OFF at or below the minimum of selected source and ON at or above the maximum of the selected source, and it will not change in between those points. Thus analog to digital interpretation has a built in hysteresis defined by minimum and maximum of the selected source, as shown in Figure 3. For example Quadrature Encoder Input signal is interpreted to be ON at or above **Maximum Range** and OFF at or below **Minimum Range**.

Control Constant Data has no unit nor minimum and maximum assigned to it, thus user has to assign appropriate constant values according to intended use.

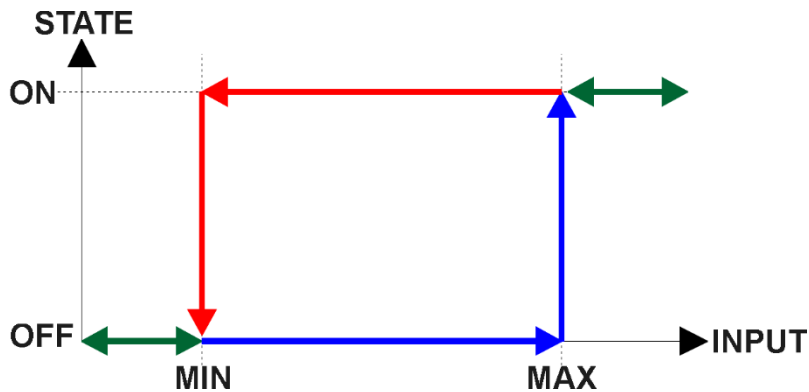


Figure 3: Analog source to Digital input

#### 1.4. Output Drive Function Blocks

The controller consists of 2 proportional truly independent outputs. Each output consists of a high side half-bridge driver able to source up to 2.5Amps. The outputs are connected to independent microcontroller timer peripherals and thus can be configured independently from 1Hz to 25kHz.

The Output Type setpoint determines what kind of signal the output produces. Changing this setpoint causes other setpoints in the group to update to match selected type. For this reason, the first setpoint that should be changed prior to configuring other setpoints is the Output Type setpoint. The supported output types by the controller are listed in Table 6 below:

Table 6: Output Type Options

0	Disabled
1	Proportional Current
2	Digital Hotshot
3	PWM Duty Cycle
4	Proportional Voltage (0-Vps)
5	Digital (0-Vps)

There are two setpoints that are associated to *Proportional Current* and *Digital Hotshot Output Types* that are not with others - these are *Dither Frequency* and *Dither Amplitude*. The dither signal is used in *Proportional Current* mode and is a low frequency signal superimposed on top of the high frequency (25kHz) signal controlling the output current. The two outputs have independent dither frequencies which can be adjusted at any time. The combination of *Dither Amplitude* and *Dither Frequency* must be appropriately selected to ensure fast response to the coil to small changes in the control inputs but not so large as to affect the accuracy or stability of the output.

In *Proportional Voltage* type, the controller measures the Vps applied to the unit and based on this information, the controller will adjust the PWM duty cycle of the signal (0-Vps amplitude) so that the average signal is the commanded target value. Thus, the output signal is not an analog one. In order to create an analog signal, a simple low pass filter can be connected externally to the controller. Note: the output signal will saturate at Vps if the **Output At Maximum Command** is set higher than the supply voltage powering the controller.

In *PWM Duty Cycle Output Type*, the controller outputs a signal (0-Vps amplitude) on a fixed output frequency set by **PWM Output Frequency** with varying PWM Duty Cycle based on

commanded input. Since both outputs are connected to independent timers, the **PWM Output Frequency** setpoint can be changed at any time for each output without affecting the other.

**Digital Output Type** offers the user with 4 different output responses as listed in Table 7. The controller will source any current required in any of the options listed in Table 7 up to 2.5Amps.

Table 7: Digital Output Responses

0	Normal On/Off
1	Inverse Logic
2	Latched Logic
3	Blinking Logic

In a *Normal* response, when the Control input commands the output ON, then the output will be turned ON. However, in an *Inverse* response, the output will be ON unless the input commands the output ON, in which case it turns OFF.

If a *Latched* response is selected, when the input commands the state from OFF to ON, the output will change state.

If a *Blinking* response is selected, then while the input command the output ON, it will blink at the rate in the **“Digital Blink Rate”** setpoint. When commanded OFF, the output will stay off. A blinking response is only available with a *Digital On/Off* type of output (not a Hotshot type.)

The *Hotshot Digital* type is different from *Digital On/Off* in that it still controls the current through the load. This type of output is used to turn on a coil then reduce the current so that the valve will remain open, as shown in Figure 4. Since less energy is used to keep the output engaged, this type of response is very useful to improve overall system efficiency. With this output type there are associated three setpoints: **Hold Current**, **Hotshot Current** and **Hotshot Time** which are used to configure form of the output signal as shown in Figure 4.

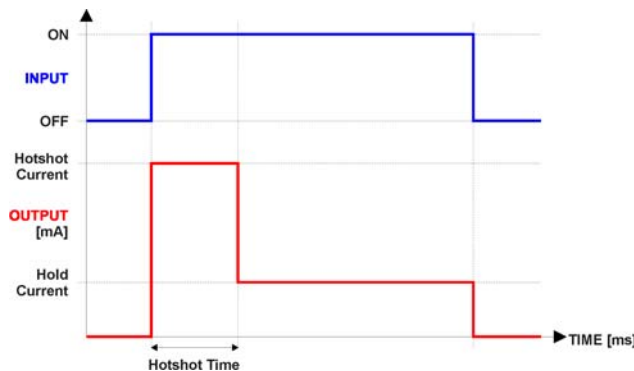


Figure 4: Hotshot Digital Profile

For Proportional outputs signal minimum and maximum values are configured with **Output At Minimum Command** and **Output At Maximum Command** setpoints. Value range for both of the setpoints is limited by selected **Output Type**.

Regardless of what type of control input is selected, the output will always respond in a linear fashion to changes in the input:

$$y = mx + a$$

$$m = \frac{Y_{max} - Y_{min}}{X_{max} - X_{min}}$$

$$a = Y_{min} - m * X_{min}$$

In the case of the Output Control Logic function block, X and Y are defined as

$X_{min}$  = Control Input Minimum ;  $Y_{min}$  = **Output at Minimum Command**

$X_{max}$  = Control Input Maximum;  $Y_{max}$  = **Output at Maximum Command**

In all cases, while X-axis has the constraint that  $X_{min} < X_{max}$ , there is no such limitation on the Y-axis. Thus configuring **Output At Minimum Command** to be greater than **Output At Maximum Command** allows output to follow control signal inversely.

In order to prevent abrupt changes at the output due to sudden changes in the command input, the user can choose to use the independent up or down ramps to smooth out the coil's response. The **Ramp Up** and **Ramp Down** setpoints are in milliseconds, and the step size of the output change will be determined by taking the absolute value of the output range and dividing it by the ramp time.

The **Control Source** setpoint together with **Control Number** setpoint determine which signal is used to drive the output. For example setting **Control Source** to *Quadrature Encoder Input Measured* and **Control Number** to (1) will connect signal measured from Quadrature Encoder Input 1 to the output in question. The input signal is scaled per input type range between 0 and 1 to form control signal. Outputs respond in a linear fashion to changes in control signal. If a non-digital signal is selected to drive digital output the command state will be 0 (OFF) at or below the "**Output At Minimum Command**", 1 (ON) at or above "**Output At Maximum Command**" and will not change in between those points.

In addition to the **Control Source** setpoint, the controller offers two more options that help increase its versatility – **Enable Source/Number/Response** and **Override Source/Number/Response** set of setpoints.

The **Enable Source** setpoint together with **Enable Number** setpoint determine the enable signal for the output in question. The **Enable Response** setpoint is used to select how output will respond to the selected Enable signal. **Enable Response** setpoint options are listed in Table 8. If a non-digital signal is selected as Enable signal the signal is interpreted as shown in Figure 3.

Table 8: Enable Response Options

0	Enable When On, Else Shutoff
1	Enable When On, Else Rampoff
2	Enable When On, Else Ramp To Max
3	Enable When On, Else Ramp To Min
4	Enable When On, Else Keep Last Value
5	Enable When Off, Else Shutoff
6	Enable When Off, Else Rampoff
7	Enable When Off, Else Ramp To Max
8	Enable When Off, Else Ramp To Min
9	Enable When Off, Else Keep Last Value

Override input allows the output drive to be configured to go to a default value in the case of the override input being engaged or disengaged, depending on the logic selected in **Override Response**, presented on Table 9. When active, the output will be driven to the value in **Output at Override Command** regardless of the value of the Control input. The **Override Source** and **Override Number** together determine the Override input signal.

Table 9: Override Response Options

0	Override When On
1	Override When Off

If a fault is detected in any of the active inputs (Control/Enable/Override) the output will respond per **Control Fault Response** setpoint as outlined in Table 10. Fault Value is defined by **Output in Fault Mode** setpoint value, which is interpreted in selected output units.

Table 10: Fault Response Options

0	Shutoff Output
1	Apply Fault Value
2	Hold Last Value

Besides Enable and Override signals controlling a particular output; another fault mode than can occur is that of a Power Supply. Power Supply fault can be enabled to detect over voltage or under voltage which will automatically disable ALL outputs. This setpoint is associated with the **Power Supply Diag** function block. Also, if the **Over Temperature Diag** function block is enabled, then a microprocessor over-temperature reading disables all the outputs until it has cooled back to the operating range.

Fault detection is available for current output types. A current feedback signal is measured and compared to desired output current value. Fault detection and associated setpoints are presented in section 1.13.

The outputs are inherently protected against a short to GND or Vps by hardware. In case of a dead short, the hardware will automatically disable the output drive, regardless of what the processor is commanding for the output. When this happens, the processor detects output hardware shutdown and commands off the output in question. It will continue to drive non-shortened outputs normally and periodically try to re-engage the short load, if still commanded to do so. If the fault has gone away

since the last time the output was engaged while shorted, the controller will automatically resume normal operation.

In the case of an open circuit, there will be no interruption of the control for any of the outputs. The processor will continue to attempt to drive the open load.

The measured current through the load is available to be broadcasted on a CAN message if desired. It is also used as the input to the diagnostic function block for each output, and an open or shorted output can be broadcasted in a DM1 message on the CAN network

## 1.5. Lookup Table Function Block

Lookup Tables are used to give an output response of up to 10 slopes per Lookup Table. There are two types of Lookup Table response based on **X-Axis Type**: *Data Response* and *Time Response* Sections 1.5.2 through 1.5.6 will describe these two **X-Axis Types** in more detail. If more than 10 slopes are required, a Programmable Logic Block can be used to combine up to three tables to get 30 slopes, as is described in Section 1.7.

There are two key setpoints that will affect this function block. The first is the **X-Axis Source** and **X-Axis Number** which together define the Control Source for the function block. When it is changed, the table is automatically updated with new defaults based on the X-Axis source selected if **Auto Update on Setpoint Changes** in the **Miscellaneous** block is *TRUE*.

### 1.5.1. Auto Update on Setpoint Changes

If **Auto Update on Setpoint Changes** is set to *TRUE*, the **X-Values** of the Lookup Tables are updated based on the **X-Axis Source** and **X-Axis Number** (i.e. the Min or Max values of the function block are updated), the associated table will also be automatically updated with default settings, based on the new X-Axis limits. However, if **Auto Update on Setpoint Changes** is set to *FALSE* then the **X-Values** will not get automatically updated on a setpoint change, but the minimum and maximum allowable ranges will be adjusted which Electronic Assistant may give warnings for out-of-range setpoints. For example, if **Quadrature Encoder Input 1** is configured with **Minimum Range: 100 RPM ( $X_{min}$ )** and **Maximum Range: 9800 RPM ( $X_{max}$ )** and is the **X-Axis Source** to **Lookup Table 1** with maximum **X Value** of 9800 RPM ( $X_{max}$ ), if **Quadrature Encoder Input 1** is then changed to **Maximum Range: 5500 RPM**, **Lookup Table 1** will not update its **X-Values** however, the maximum command ( $X_{max}$ ) will be set to 5500 RPM so editing the maximum **X Value** to something higher than 5500 RPM Electronic Assistant will display a warning. It is up to the user's discretion to appropriately select the values in the Lookup Tables when **Auto Update on Setpoint Changes** is set to *FALSE*.

### 1.5.2. X-Axis, Input Data Response

In the case where the **X-Axis Type** = *Data Response*, the points on the X-Axis represents the data of the control source. These values must be selected within the range of the control source.

When selecting X-Axis data values, there are no constraints on the value that can be entered into any of the X-Axis points. The user should enter values in increasing order to be able to utilize the entire table. Therefore, when adjusting the X-Axis data, it is recommended that  $X_{10}$  is changed first, then lower indexes in descending order as to maintain the below:

$$X_{min} \leq X_0 \leq X_1 \leq X_2 \leq X_3 \leq X_4 \leq X_5 \leq X_6 \leq X_7 \leq X_8 \leq X_9 \leq X_{10} \leq X_{max}$$

As stated earlier,  $X_{min}$  and  $X_{max}$  will be determined by the X-Axis Source that has been selected.

If some of the data points are 'Ignored' as described in Section 1.5.4, they will not be used in the X-Axis calculation shown above. For example, if points  $X_4$  and higher are ignored, the formula becomes  $X_{min} \leq X_0 \leq X_1 \leq X_2 \leq X_3 \leq X_{max}$  instead.

### 1.5.3. Y-Axis, Lookup Table Output

The Y-Axis has no constraints on the data that it represents. This means that inverse, or increasing/decreasing or other responses can be easily established.

In all cases, the controller looks at the **entire range** of the data in the Y-Axis setpoints, and selects the lowest value as the  $Y_{min}$  and the highest value as the  $Y_{max}$ . They are passed directly to other function blocks as the limits on the Lookup Table output. (i.e used as  $X_{min}$  and  $X_{max}$  values in linear calculations.)

However, if some of the data points are 'Ignored' as described in Section 1.5.4, they will not be used in the Y-Axis range determination. Only the Y-Axis values shown on EA will be considered when establishing the limits of the table when it is used to drive another function block, such as a Math Function Block.

### 1.5.4. Default Configuration, Data Response

By default, all Lookup Tables in the ECU are disabled (**X-Axis Source** equals *Control Not Used*). Lookup Tables can be used to create the desired response profiles. If a Universal Input is used as the X-Axis, the output of the Lookup Table will be what the user enters in **Y-Values** setpoints.

Recall, any controlled function block which uses the Lookup Table as an input source will also apply a linearization to the data. **Therefore, for a 1:1 control response, ensure that the minimum and maximum values of the output correspond to the minimum and maximum values of the table's Y-Axis.**

All tables (1 to 2) are disabled by default (no control source selected). However, should an **X-Axis Source** be selected, the **Y-Values** defaults will be in the range of 0 to 100% as described in the "Y-Axis, Lookup Table Output" section above. X-Axis minimum and maximum defaults will be set as described in the "X-Axis, Data Response" section above.

**By default, the X and Y axes data is setup for an equal value between each point from the minimum to maximum in each case.**

### 1.5.5. Point To Point Response

By default, the X and Y axes are setup for a linear response from point (0,0) to (10,10), where the output will use linearization between each point, as shown in Figure 4. To get the linearization, each "**Point N – Response**", where N = 1 to 10, is setup for a 'Ramp To' output response.

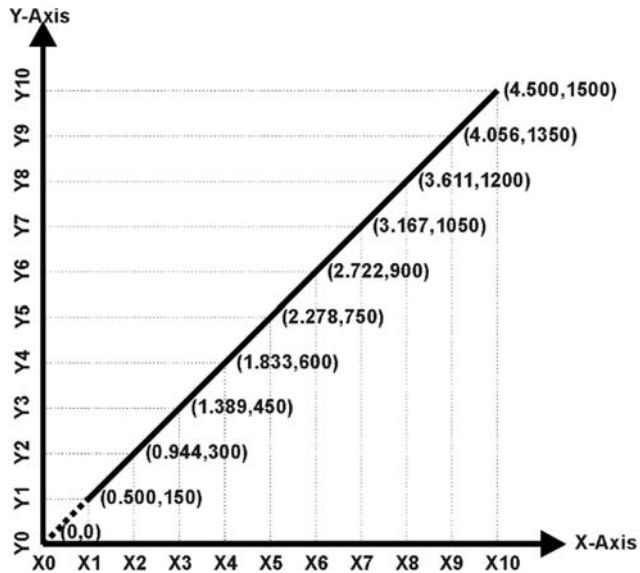


Figure 5: Lookup Table with "Ramp To" Data Response

Alternatively, the user could select a 'Jump To' response for "Point N – Response", where N = 1 to 10. In this case, any input value between  $X_{N-1}$  to  $X_N$  will result in an output from the Lookup Table function block of  $Y_N$ .

An example of a Math function block (0 to 100) used to control a default table (0 to 100) but with a 'Jump To' response instead of the default 'Ramp To' is shown in Figure 5.

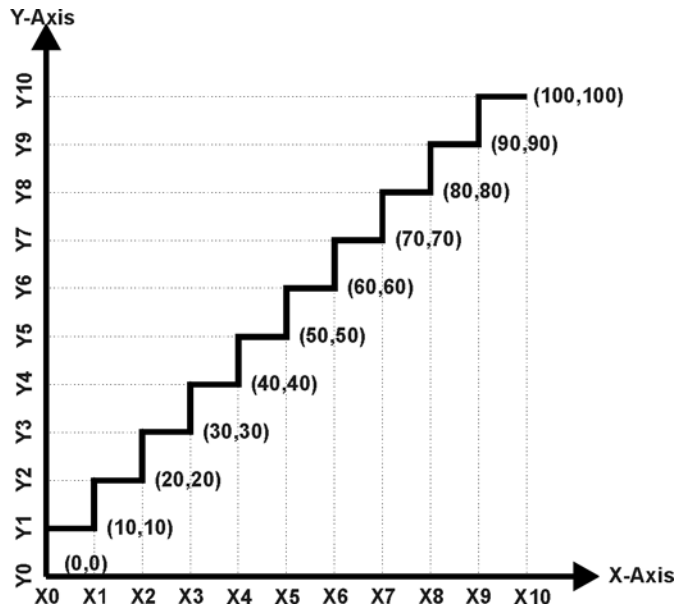


Figure 6: Lookup Table with "Jump To" Data Response

Lastly, any point except (0,0) can be selected for an 'Ignore' response. If "Point N – Response" is set to ignore, then all points from  $(X_N, Y_N)$  to  $(X_{10}, Y_{10})$  will also be ignored. For all data greater than  $X_{N-1}$ , the output from the Lookup Table function block will be  $Y_{N-1}$ .

A combination of *Ramp To*, *Jump To* and *Ignore* responses can be used to create an application specific output profile.

### 1.5.6. X-Axis, Time Response

As mentioned in Section 1.5, a Lookup Table can also be used to get a custom output response where the **X-Axis Type** is a *'Time Response.'* When this is selected, the X-Axis now represents time, in units of milliseconds, while the Y-Axis still represents the output of the function block. There is also another setpoint associated to the Lookup Table when configured to *Time Response* which is the **Table Auto-Cycle** setpoint.

In this case, the **X-Axis Source** is treated as a digital input. If the signal is actually an analog input, it is interpreted like a digital input per Figure 3. When the control input is ON, the output will be changed over a period of time based on the profile in the Lookup Table. There are two different scenarios on how the Lookup Table will react once the profile is finished. The first option is when **Table Auto-Cycle** is set to *FALSE* in which case, once the profile has finished (i.e. index 10, or *Ignored* response), the output will remain at the last output at the end of the profile until the control input turns OFF. The second option is when **Table Auto-Cycle** is set to *TRUE* in which case, once the profile has finished (i.e. index 10, or *Ignored* response), the Lookup Table will automatically return to the 1<sup>st</sup> response and will continually be auto-cycling for as long as the input remains in the ON state.

When the control input is OFF, the output is always at zero. When the input comes ON, the profile ALWAYS starts at position (X<sub>0</sub>, Y<sub>0</sub>) which is 0 output for 0ms.

In a time response, the interval time between each point on the X-axis can be set anywhere from 1ms to 1min. [60,000 ms]

### 1.6. Math Function Block

There are two mathematical function blocks that allow the user to define basic algorithms. A math function block can take up to five input signals. Each input is then scaled according to the associated limit and scaling setpoints.

Inputs are converted into percentage value based on the "Function X Input Y Minimum" and "Function X Input Y Maximum" values selected. For additional control the user can also adjust the "Function X Input Y Scaler". By default, each input has a scaling 'weight' of 1.0 However, each input can be scaled from -1.0 to 1.0 as necessary before it is applied in the function.

For example, in the case where the user may want to combine two inputs such that a quadrature Encoder speed (Input 1) is the primary control of an output, but the speed can be incremented or decremented based on the quadrature Encoder step count (Input 2), it may be desired that 75% of the scale is controlled by the Encoder speed, while the current step count can increase or decrease the min/max output by up to 25%. In this case, Input 1 would be scaled with 0.75, while Input 2 uses 0.25. The resulting addition will give a command from 0 to 100% based on the combined positions of both inputs.

A mathematical function block includes four selectable functions, which each implements equation  $A \text{ operator } B$ , where A and B are function inputs and operator is function selected with setpoint **Math function X Operator**. Setpoint options are presented in the table below. The functions are connected together, so that result of the preceding function goes into Input A of the next function. Thus Function 1 has both Input A and Input B selectable with setpoints, where Functions 2 to 3 have only Input B selectable. Input is selected by setting "**Function X Input Y Source**" and

**Function X Input Y Number.** If **Function X Input B Source** is set to 0 'Control not used' signal goes through function unchanged.

$$\text{Math Block Output} = \left( \left( (\text{InA } op1 \text{ InB1}) \right) op2 \text{ InB2} \right) op3 \text{ InB3} )$$

*Table 11: Math Function Operators*

0	=	True when InA Equals InB
1	!=	True when InA Not Equal InB
2	>	True when InA Greater Than InB
3	>=	True when InA Greater Than or Equal InB
4	<	True when InA Less Than InB
5	<=	True when InA Less Than or Equal InB
6	OR	True when InA or InB is True
7	AND	True when InA and InB are True
8	XOR	True when InA/InB is True, but not both
9	+	Result = InA plus InB
10	-	Result = InA minus InB
11	x	Result = InA times InB
12	/	Result = InA divided by InB
13	MIN	Result = Smallest of InA and InB
14	MAX	Result = Largest of InA and InB
15	MAX-MIN	Result = Largest minus Smallest of InA and InB

For logic operations (6, 7, 8) scaled input greater or equal to 1 is treated as TRUE. For logic operations (0 to 8), the result of the function will always be 0 (FALSE) or 1 (TRUE). For the arithmetic functions (9 to 15), it is recommended to scale the data such that the resulting operation will not exceed full scale (0 to 100%) and saturate the output result.

When dividing, a zero divider will always result in a 100% output value for the associated function.

Lastly the resulting mathematical calculation, presented as a percentage value, can be scaled into the appropriate physical units using the **Math Output Minimum Range** and **Math Output Maximum Range** setpoints. These values are also used as the limits when the Math Function I selected as the input source for another function block.

## 1.7. Programmable Logic Function Block

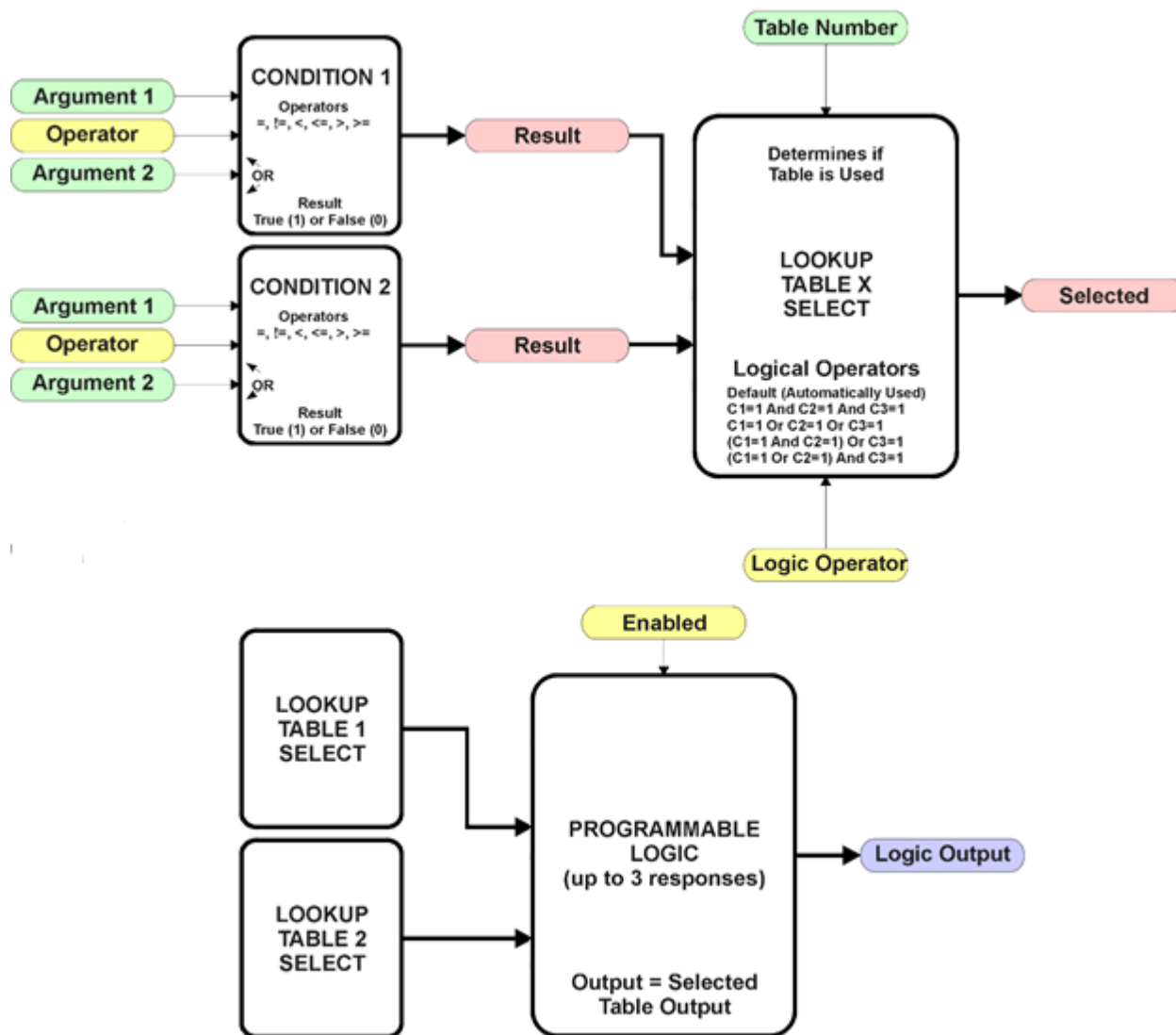


Figure 7: Programmable Logic Function Block

This function block is obviously the most complicated of them all, but very powerful. The Programmable Logic can be linked to up to two tables, either one of which would be selected only under given conditions. Either one of the tables can be associated with the logic, and which ones are used is fully configurable.

Should the conditions be such that a particular table (1 or 2) has been selected as described in Section 1.5.2, then the output from the selected table, at any given time, will be passed directly to the Logic Output.

Therefore, up to two different responses to the same input, or two different responses to different inputs, can become the input to another function block, such as an Output X Drive. To do this, the **Control Source** for the reactive block would be selected to be the 'Programmable Logic Function Block.'

In order to enable the Programmable Logic block, the **Programmable Logic Block Enabled** setpoint must be set to True. The block is disabled by default.

Logic is evaluated in the order shown in Figure 8. Only if a lower number table has not been selected will the conditions for the next table be looked at. **The default table is always selected as soon as it is evaluated. It is therefore required that the default table always be the highest number in any configuration.**

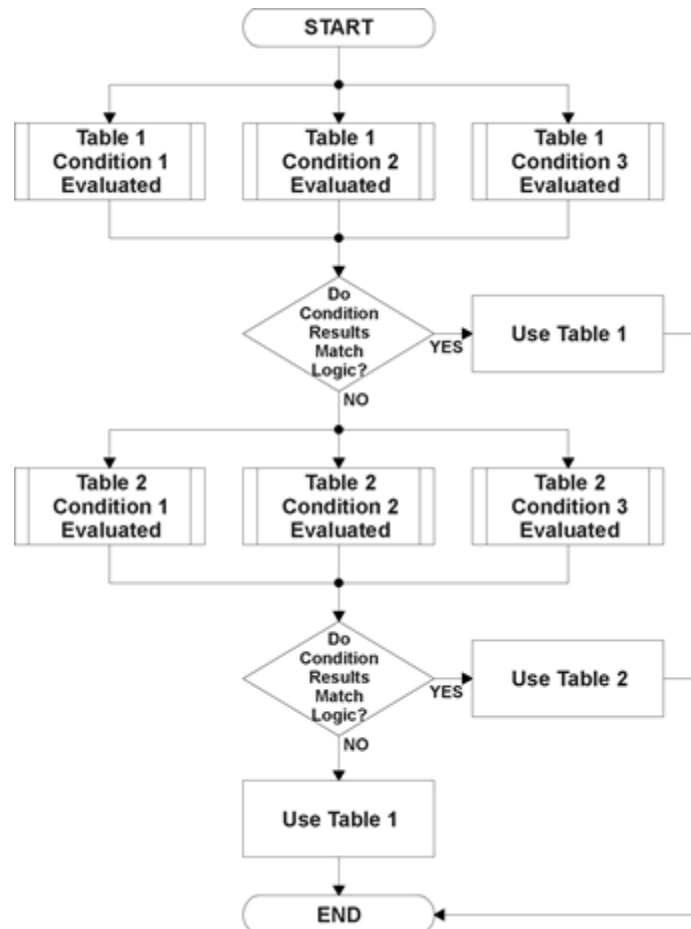


Figure 8: Programmable Logic Flowchart

### 1.7.1. Conditions Evaluation

The first step in determining which table will be selected as the active table is to first evaluate the conditions associated with a given table. Each table has associated with it up to three conditions that can be evaluated.

Argument 1 is always a logical output from another function block. As always, the source is a combination of the functional block type and number, setpoints **Table X, Condition Y, Argument 1 Source** and **Table X, Condition Y, Argument 1 Number**, where both X = 1 to 3 and Y = 1 to 3.

Argument 2 on the other hand, could either be another logical output such as with Argument 1, OR a constant value set by the user. To use a constant as the second argument in the operation, set **Table X, Condition Y, Argument 2 Source** to *Control Constant Data*. Note that the constant value has no unit associated with it in EA, so the user must set it as needed for the application.

The condition is evaluated based on the **Table X, Condition Y Operator** selected by the user. It is always = ,*Equal*' by default. The only way to change this is to have two valid arguments selected for any given condition. Options for the operator are listed in Table 12.

Table 12: Condition Operator Options

0	=, <i>Equal</i>
1	!=, <i>Not Equal</i>
2	>, <i>Greater Than</i>
3	>=, <i>Greater Than or Equal</i>
4	<, <i>Less Than</i>
5	<=, <i>Less Than or Equal</i>

By default, both arguments are set to '*Control Source Not Used*' which disables the condition, and automatically results in a value of N/A as the result. Although Figure 8 shows only True or False as a result of a condition evaluation, the reality is that there could be four possible results, as described in Table 13.

Table 13: Condition Evaluation Results

Value	Meaning	Reason
0	False	(Argument 1) Operator (Argument 2) = False
1	True	(Argument 1) Operator (Argument 2) = True
2	Error	Argument 1 or 2 output was reported as being in an error state
3	Not Applicable	Argument 1 or 2 is not available (i.e. set to ' <i>Control Source Not Used</i> ')

### 1.7.2. Table Selection

In order to determine if a particular table will be selected, logical operations are performed on the results of the conditions as determined by the logic in Section 1.7.1. There are several logical combinations that can be selected, as listed in Table 14.

Table 14: Conditions' Logical Operator Options

0	<i>Default Table</i>
1	<i>Cnd1 And Cnd2 And Cnd3</i>
2	<i>Cnd1 Or Cnd2 Or Cnd3</i>
3	<i>(Cnd1 And Cnd2) Or Cnd3</i>
4	<i>(Cnd1 Or Cnd2) And Cnd3</i>

Not every evaluation is going to need all three conditions. The case given in the earlier section, for example, only has one condition listed, i.e. that the Engine RPM be below a certain value. Therefore, it is important to understand how the logical operators would evaluate an Error or N/A result for a condition.

Table 15: Conditions Evaluation Based on Selected Logical Operator

Logical Operator	Select Conditions Criteria
Default Table	Associated table is automatically selected as soon as it is evaluated.
Cnd1 And Cnd2 And Cnd3	<p><b>Should be used when two or three conditions are relevant, and all must be true to select the table.</b></p> <p>If any condition equals False or Error, the table is not selected. An N/A is treated like a True. If all three conditions are True (or N/A), the table is selected.</p>

	If((Cnd1==True) &&(Cnd2==True)&&(Cnd3==True)) Then Use Table
Cnd1 Or Cnd2 Or Cnd3	<b>Should be used when only one condition is relevant. Can also be used with two or three relevant conditions.</b>  If any condition is evaluated as True, the table is selected. Error or N/A results are treated as False  If((Cnd1==True)    (Cnd2==True)    (Cnd3==True)) Then Use Table
(Cnd1 And Cnd2) Or Cnd3	<b>To be used only when all three conditions are relevant.</b>  If both Condition 1 and Condition 2 are True, OR Condition 3 is True, the table is selected. Error or N/A results are treated as False  If( ((Cnd1==True)&&(Cnd2==True))    (Cnd3==True) ) Then Use Table
(Cnd1 Or Cnd2) And Cnd3	<b>To be used only when all three conditions are relevant.</b>  If Condition 1 And Condition 3 are True, OR Condition 2 And Condition 3 are True, the table is selected. Error or N/A results are treated as False  If( ((Cnd1==True)  ((Cnd2==True)) && (Cnd3==True) ) Then Use Table

The default **Table X, Conditions Logical Operator** for Table 1 and Table 2 is *Cnd1 And Cnd2 And Cnd3*, while Table 3 is set to be the *Default Table*.

### 1.7.3. Logic Block Output

Recall that Table X, where X = 1 to 3 in the Programmable Logic function block does NOT mean Lookup Table 1 to 3. Each table has a setpoint **Table X – Lookup Table Block Number** which allows the user to select which Lookup Tables they want associated with a particular Programmable Logic Block. The default tables associated with each logic block are listed in Table 8.

Table 16: Programmable Logic Block Default Lookup Tables

Programmable Logic Block Number	Table 1 – Lookup Table Block Number	Table 2 – Lookup Table Block Number	Table 3 – Lookup Table Block Number
1	1	2	3

If the associated Lookup Table does not have an **X-Axis Source** selected, then the output of the Programmable Logic block will always be “Not Available” so long as that table is selected. However, should the Lookup Table be configured for a valid response to an input, be it Data or Time, the output of the Lookup Table function block (i.e. the Y-Axis data that has been selected based on the X-Axis value) will become the output of the Programmable Logic function block so long as that table is selected.

Unlike all other function blocks, the Programmable Logic does NOT perform any linearization calculations between the input and the output data. Instead, it mirrors exactly the input (Lookup Table) data. Therefore, when using the Programmable Logic as a control source for another function block, it is HIGHLY recommended that all the associated Lookup Table Y-Axes either be (a) Set between the 0 to 100% output range or (b) all set to the same scale.

## 1.8. PID Control Function Block

The PID Control function block is an independent logic block, but it is normally intended to be associated with proportional output control blocks described earlier. When the **Control Source** for an output has been setup as a 'PID Function Block', the command from the selected PID block drives the physical output on the High Temperature Quadrature Encoder Input 2 Output Valve Controller.

The **PID Target Command Source** and **PID Target Command Number** setpoints determine control input and the **PID Feedback Input Source** and **PID Feedback Input Number** setpoints determine the established the feedback signal to the PID function block. The **PID Response Profile** will use the selected inputs as per the options listed in Table 1. When active, the PID algorithm will be called every **PID Loop Update Rate** in milliseconds.

Table 17: PID Response Options

0	Single Output
1	Setpoint Control
2	On When Over Target
3	On When Below Target

When a '*Single Output*' response is selected, the Target and Feedback inputs do not have to share the same units. In both cases, the signals are converted to a percentage values based on the minimum and maximum values associated with the source function block.

For example, a CAN command could be used to set the target value, in which case it would be converted to a percentage value using **Receive Data Min** and **Receive Data Max** setpoints in the appropriate *CAN Receive X* function block. The closed-loop feedback signal (i.e. a digital quadrature decoder input) could be connected to '*Quadrature Encoder Input 1*' and selected as the feedback source. In this case the value of the input would be converted to a percentage based on the **Minimum Range** and **Maximum Range** setpoints in the input block. The output of the PID function would depend on the difference between the commanded target and the measured feedback as a percentage of each signals range. In this mode, the output of the block would be a value from -100% to 100%.

When a *Setpoint Control* response is selected, the **PID Target Command Source** automatically gets updated to *Control Constant Data* and cannot be changed. The value set in the associated constant in the Constant Data List function block becomes the desired target value. In this case, both the target and the feedback values are assumed to be in same units and range. The minimum and maximum values for the feedback automatically become the constraints on the constant target. In this mode, the output of the block would be a value from 0% to 100%.

For example, if the feedback was setup as a Quadrature Encoder Step Count input (with default range of -60000...60000 steps), a **Constant Value X** setpoint set to 40000 would automatically be converted to 83.33%. The PID function would adjust the output as needs to have the measured feedback to maintain that target value.

The last two response options, '*On When Over Target*' and '*On When Under Target*', are designed to allow the user to combine the two proportional outputs as a push-pull drive for a system. Both

outputs must be setup to use the same control input (linear response) and feedback signal in order to get the expected output response. In this mode, the output would be between 0% to 100%.

In Order to allow the output to stabilize, the user can select a non-zero value for **PID Delta Tolerance**. If the absolute value of  $Error_k$  is less than this value,  $Error_k$  in the formula below will be set to zero.

The PID algorithm used is shown below, where  $G$ ,  $K_i$ ,  $T_i$ ,  $K_d$ ,  $T_d$  and  $Loop\_Update\_Rate$  are configurable parameters.

$$PIDOutput_k = P_k + I_k + D_k$$

$$P_k = P\_Gain * Error_k$$

$$I_k = I\_Gain * ErrorSum_k$$

$$D_k = D\_Gain * (Error_k - Error_{k-1})$$

$$Error_k = Target - Feedback$$

$$ErrorSum_k = ErrorSum_{k-1} + Error_k$$

$$P\_Gain = G$$

$$I\_Gain = G * K_i * T / T_i \text{ (Note: If } T_i \text{ is zero, } I\_Gain = 0)$$

$$D\_Gain = G * K_d * T_d / T$$

$$T = Loop\_Update\_Rate * 0.001$$

Each system will have to be turned for the optimum output response. Response times, overshoots and other variables will have to be decided by the customer using an appropriate PID tuning strategy. Axiomatic is not responsible for tuning the control system.

### 1.9. Set / Reset Function Block

**Set-Reset Block** consists of only 2 control sources: **Reset Source** and **Set Source**. The purpose of these blocks is to simulate a modified latching function in which the 'Reset Signal' has more precedence. The 'latching' function works as per the Table 18 below.

Table 18: Set-Reset Function Block Operation

'Set Signal'	'Reset Signal'	'Set-Reset Block Output' (Initial State: OFF)
OFF	OFF	Latched State
OFF	ON	OFF
ON	OFF	ON
ON	ON	OFF

If either **Reset Source** or **Set Source** is a non-digital signal, the signal is interpreted as OFF if the signal is at or below the minimum of the selected source; and is interpreted as ON at or above the maximum of the selected source. The ON/OFF states of the signal will not change in between.

Thus analog to digital interpretation has a built-in hysteresis defined by the signal's minimum and maximum ranges, as shown in Figure 3.

An example of the explanation above: if Quadrature Encoder Input 1 is configured as a Speed input with **Minimum Range**: 500 RPM and **Maximum Range**: 3500 RPM and is used in this block as **Set Source**. If the measured input signal is 500 RPM, the 'Set Signal' will be OFF and will remain OFF until the measured input signal has reached 3500 RPM or higher (but less than **Maximum Error**) in which case 'Set Signal' will switch to ON. At this point, the state of 'Set Signal' will remain ON until the measured input signal has reached 500 RPM (but higher than **Minimum Error**) in which case the state of 'Set Signal' will switch OFF.

As seen in Table 18 above, the 'Reset Signal' has more precedence over the 'Set Signal' - as long as the state of 'Reset Signal' is ON, the state of 'Set-Reset Block Output' will be OFF. In order to create an ON state in 'Set-Reset Block Output' the state of 'Reset Signal' must be OFF while the state of 'Set Signal' is ON. In this case, the state of 'Set-Reset Block Output' will remain ON even if 'Set Signal' turns OFF as long as 'Reset Signal' remains OFF. As soon as the 'Reset Signal' turns ON the 'Set-Reset Block Output' will turn OFF regardless of the state of 'Set Signal'.

### 1.10. Diagnostic Trouble Code (DTC) React

The DTC React function block will allow a received DTC sent from another ECU on a DM1 message to be used as an input to any other function block in order to disable an output, for example. Up to three SPN/FMI combinations can be selected.

Should a DM1 message be received with the SPN/FMI combination defined, the corresponding DTC State will be set to ON. Once ON, if the same SPN/FMI combination has not been received again after 3 seconds, the DTC State will be reset to OFF.

The DTC could be used as a digital input for any function block as appropriate.

## 1.11. CAN Transmit Message Function Block

The CAN Transmit function block is used to send any output from another function block (i.e. Quadrature Encoder Input, CAN Received) to the J1939 network. The AX023240 ECU has four CAN Transmit Messages with each having one user-defined signal.

### 1.11.1. CAN Transmit Message Setpoints

The **Transmit PGN** setpoint sets PGN used with the message. User should be familiar with the SAE J1939 standard, and select values for PGN/SPN combinations as appropriate from section J1939/71.

**Repetition Rate** setpoint defines the interval used to send the message to the J1939 network. If the **Repetition Rate** is set to zero, the message is disabled unless it shares its PGN with another message. In case of a shared PGN repetition rate of the LOWEST numbered message are used to send the message 'bundle'.

By default, all messages are sent on Proprietary B PGNs as broadcast messages. Thus **Transmit Message Priority** is always initialized to 6 (low priority) and the **Destination Address** setpoint is not used. This setpoint is only valid when a PDU1 PGN has been selected, and it can be set either to the Global Address (0xFF) for broadcasts, or sent to a specific address as setup by the user.

### 1.11.2. CAN Transmit Signal Setpoints

Each CAN transmit message has one associated signal, which define data inside the Transmit message. **Control Source** setpoint together with **Control Number** setpoint define the signal source of the message. **Control Source** and **Control Number** options are listed in Table 4 and Table 5. Setting **Control Source** to *Control Not Used* disables the signal.

**Transmit Data Size** setpoint determines how many bits signal reserves from the message. **Transmit Data Index in Array** determines in which of 8 bytes of the CAN message LSB of the signal is located. Similarly **Transmit Bit Index in Byte** determines in which of 8 bits of a byte the LSB is located. These setpoints are freely configurable, thus **it is the user's responsibility to ensure that signals do not overlap and mask each other.**

**Transmit Data Resolution** setpoint determines the scaling done on the signal data before it is sent to the bus. **Transmit Data Offset** setpoint determines the value that is subtracted from the signal data before it is scaled. Offset and Resolution are interpreted in units of the selected source signal.

## 1.12. CAN Receive Function Block

The CAN Receive function block is designed to take any SPN from the J1939 network, and use it as an input to another function block.

The **Receive Message Enabled** is the most important setpoint associated with this function block and it should be selected first. Changing it will result in other setpoints being enabled/disabled as appropriate. By default ALL receive messages are disabled.

Once a message has been enabled, a Lost Communication fault will be flagged if that message is not received within the **Receive Message Timeout** period. This could trigger a Lost Communication event as described in section 1.13. In order to avoid timeouts on a heavily saturated network, it is recommended to set the period at least three times longer than the expected update rate. To disable the timeout feature, simply set this value to zero, in which case the received message will never trigger a Lost Communication fault.

By default, all control messages are expected to be sent to the High Temperature Quadrature Encoder Input 2 Output Valve Controller on Proprietary B PGNs. However, should a PDU1 message be selected, the High Temperature 2 Input 2 Output Valve Controller can be setup to receive it from any ECU by setting the **Specific Address that sends the PGN** to the Global Address (0xFF). If a specific address is selected instead, then any other ECU data on the PGN will be ignored.

The **Receive Data Size**, **Receive Data Index in Array (LSB)**, **Receive Bit Index in Byte (LSB)**, **Receive Resolution** and **Receive Offset** can all be used to map any SPN supported by the J1939 standard to the output data of the Received function block.

As mentioned earlier, a CAN receive function clock can be selected as the source of the control input for the output function blocks. When this is the case, the **Received Data Min** (Off Threshold) and **Received Data Max** (On Threshold) setpoints determine the minimum and maximum values of the control signal. As the names imply, they are also used as the On/Off thresholds for digital output types. These values are in whatever units the data is AFTER the resolution and offset is applied to CAN receive signal.

The High Temperature Quadrature Encoder Input 2 Output Valve Controller I/O supports up to four unique CAN Receive Messages.

### 1.13. Diagnostic Function Blocks

The High Temperature Quadrature Encoder Input 2 Output Valve Controller supports diagnostic messaging. DM1 message is a message, containing Active Diagnostic Trouble Codes (DTC) that is sent to the J1939 network in case a fault has been detected. A Diagnostic Trouble Code is defined by the J1939 standard as a four byte value.

In addition to supporting the DM1 message, the following are supported:

DM2	Previously Active Diagnostic Trouble Codes	Sent only on request
SPN	Suspect Parameter Number	(user defined)
FMI	Failure Mode Identifier	(see Table 21 and Table 22)
CM	Conversion Method	(always set to 0)
OC	Occurrence Count	(number of times the fault has happened)
DM3	Diagnostic Data Clear/Reset of Previously Active DTCs	Done only on request
DM11	Diagnostic Data Clear/Reset for Active DTCs	Done only on request

Fault detection and reaction is associated with the two Quadrature Encoder Inputs and with the two Proportional Outputs. However not all the input and output types support fault diagnostics. Fault diagnostics are not available for digital input types, and thus diagnostic setpoints are not

used with them. The Proportional Outputs are associated with current feedback, which is utilized with 'Proportional Current' and 'Digital Hotshot' output types. For other output types, the output fault detection/reaction setpoints are ignored. In addition to input/output faults, the High Temperature Quadrature Decoder Input 2 Output Valve Controller can also detect/react to three additional faults namely power supply fault, over temperature fault and communication fault.

The EA provides several setpoints to configure diagnostics. Input and output error diagnostic setpoints are among the setpoint group of each input/output and diagnostic setpoints for additional faults are presented as their own setpoint groups in the EA.

Fault detection thresholds are presented in Table 19. Input errors can be flagged as either a high or low occurrence, thus there are two user selectable threshold value setpoints **Maximum error** and **Minimum error**. Input error thresholds are interpreted in **Input Sensor Type** units. Changing input type will change **Minimum error** and **Maximum error** to corresponding default values, thus Input Sensor Type should be set before adjusting **Minimum error** and **Maximum error** setpoints. Fault detection can be performed only if the thresholds are within the range of permitted values which are listed in Table 19.

Power Supply fault can be also flagged as either a high or low occurrence and has two selectable threshold setpoints. Over Temperature fault reacts only to a single condition and thus, the only one threshold setpoint is supplied. Lost Communication fault occurs if no CAN messages are received within **Receive Message Timeout** time. The proportional output can be selected to disable in a case of a power supply and/or temperature error, by setting **Power Fault Disables Outputs** and/or **Over Temperature Shutdown** setpoint value to 'True'.

A proportional output fault is monitored from measured current feedback and is thus applicable only on current output types (i.e. Proportional Current and Digital Hotshot). Measured current feedback value is compared with desired current value and if the difference between the two is greater than **Hysteresis to Clear Fault** setpoint value, an open circuit will be flagged. A hardware shutdown will occur if the output is sourcing greater than 2.5A +/- 0.5A, most likely due to a short circuit on the load. Output diagnostics are not available for non-current output types.

Table 19: Fault Detect Thresholds

Fault	Minimum Threshold	Maximum Threshold
Universal Input	Minimum Error	Maximum Error
Proportional Output	Hysteresis to Clear Fault	~2.5A * (Hardware)
Power Supply	Power Undervoltage Threshold	Power Overvoltage Threshold
Over Temperature	N/A	Over Temperature Threshold
Lost Communication	N/A	Received Message Timeout

A hysteresis can be applied to prevent rapid setting and clearing of the error flag when signal value is near the fault detection threshold. Input error and additional error detection hysteresis is configured with **Hysteresis to clear fault** setpoint.

**Generate Diagnostic Messages** setpoint determines whether an active fault generates diagnostic trouble code (DTC) that is sent to J1939 network as part of diagnostic message (DM). So long as even one Diagnostic function block has **Generate Diagnostic Messages** set to *True*, the High Temperature Quadrature Encoder Input 2 Output Valve Controller will send the DM1 message every one second, regardless of whether or not there are any active faults, as recommended by standard. While there are no active DTCs, the High Temperature Quadrature Encoder Input 2

Output Valve Controller will send “No Active Faults” message. If a previously inactive DTC becomes active, a DM1 will be sent immediately to reflect this. As soon as the last active DTC goes inactive, a DM1 indicating that there are no more active DTCs will be sent.

If there is more than one active DTC at any given time, the regular DM1 message will be sent using a multipacket message to the Requester Address using the Transport Protocol (TP).

When the fault is linked to a DTC, a non-volatile log of the occurrence count (OC) is kept. As soon as the controller detects a new (previously inactive) fault, it will start decrementing the **Delay Before Sending DM1** timer for that Diagnostic function block. If the fault has remained present during the delay time, then the controller will set the DTC to active, and will increment the OC in the log. A DM1 will immediately be generated that includes the new DTC. The timer is provided so that intermittent faults do not overwhelm the network as the fault comes and goes, since a DM1 message would be sent every time the fault shows up or goes away.

By default, the fault flag is cleared when error condition that has caused it goes away. The DTC is made Previously Active and is it is no longer included in the DM1 message. To identify a fault having happened, even if the condition that has caused is one away, the **Event Cleared only by DM11** setpoint can be set to *True*. This configuration enables DTC to stay Active, even after the fault flag has been cleared, and be included in DM1 message until a Diagnostic Data Clear/Reset for Active DTCs (DM11) has been requested.

As defined by J1939 Standard the first byte of the DM1 message reflects the Lamp status. “**Lamp Set by Event in DM1**” setpoint determines the lamp type set in this byte of DTC. “**Lamp Set by Event in DM1**” setpoint options are listed in Table 20. By default, the ‘*Amber, Warning*’ lamp is typically the one set be any active fault.

Table 20: Lamp Set by Event in DM1 Options

0	Protect
1	Amber Warning
2	Red Stop
3	Malfunction

**SPN for Event used in DTC** defines suspect parameter number used as part of DTC. The default value zero is not allowed by the standard, thus no DM will be sent unless **SPN for Event used in DTC** in is configured to be different from zero. **It is user’s responsibility to select SPN that will not violate J1939 standard.** When the **SPN for Event used in DTC** is changed, the OC of the associated error log is automatically reset to zero.

Table 21: FMI for Event Used in DTC Options

0	Data Valid But Above Normal Operational Range - Most Severe Level
1	Data Valid But Below Normal Operational Range - Most Severe Level
2	Data Intermittent
3	Voltage Above Normal, Or Shorted To High Source
4	Voltage Below Normal, Or Shorted To Low Source
5	Current Below Normal Or Open Circuit
6	Current Above Normal Or Grounded Circuit
7	Mechanical Error
8	Abnormal Frequency Or Pulse Width Or Period
9	Abnormal Update Rate
10	Abnormal Rate Of Change
11	Root Cause Not Known
12	Bad Component
13	Out Of Calibration
14	Special Instructions
15	Data Valid But Above Normal Operating Range – Least Severe Level
16	Data Valid But Above Normal Operating Range – Moderately Severe Level
17	Data Valid But Below Normal Operating Range – Least Severe Level
18	Data Valid But Below Normal Operating Range – Moderately Severe Level
19	Network Error
20	Data Drifted High
21	Data Drifted Low
31	Condition Exists

Every fault has associated a default FMI with them. The used FMI can be configured with **FMI for Event Used in DTC** setpoint, presented in Table 21. When FMI is selected from Low Fault FMIs in Table 22 for a fault that can be flagged either high or low occurrence, the high occurrence automatically uses corresponding High Fault FMI by Table 22. If any other FMI is selected than the Low Fault FMI from the table ref, then both the low and high fault will be assigned the same FMI.

Table 22: Low Fault FMIs and corresponding High Fault FMIs

Low Fault FMIs	High Fault FMIs
FMI=1, Data Valid But Below Normal Operation Range – Most Severe Level	FMI=0, Data Valid But Above Normal Operational Range – Most Severe Level
FMI=4, Voltage Below Normal, Or Shorted to Low Source	FMI=3, Voltage Above Normal, Or Shorted To High Source
FMI=5, Current Below Normal Or Open Circuit	FMI=6, Current Above Normal Or Grounded Circuit
FMI=17, Data Valid But Below Normal Operating Range – Least Severe Level	FMI=15, Data Valid But Above Normal Operating Range – Least Severe Level
FMI=18, Data Valid But Below Normal Operating Level – Moderately Severe Level	FMI=16, Data Valid But Above Normal Operating Range – Moderately Severe Level
FMI=21, Data Drifted Low	FMI=20, Data Drifted High

## 2. Installation Instructions

### 2.1. Dimensions and Pinout

The High Temperature Quadrature Encoder Input 2 Output Valve Controller is packaged in a plastic housing from Deutsch IPD. The assembly carries an IP67 rating.

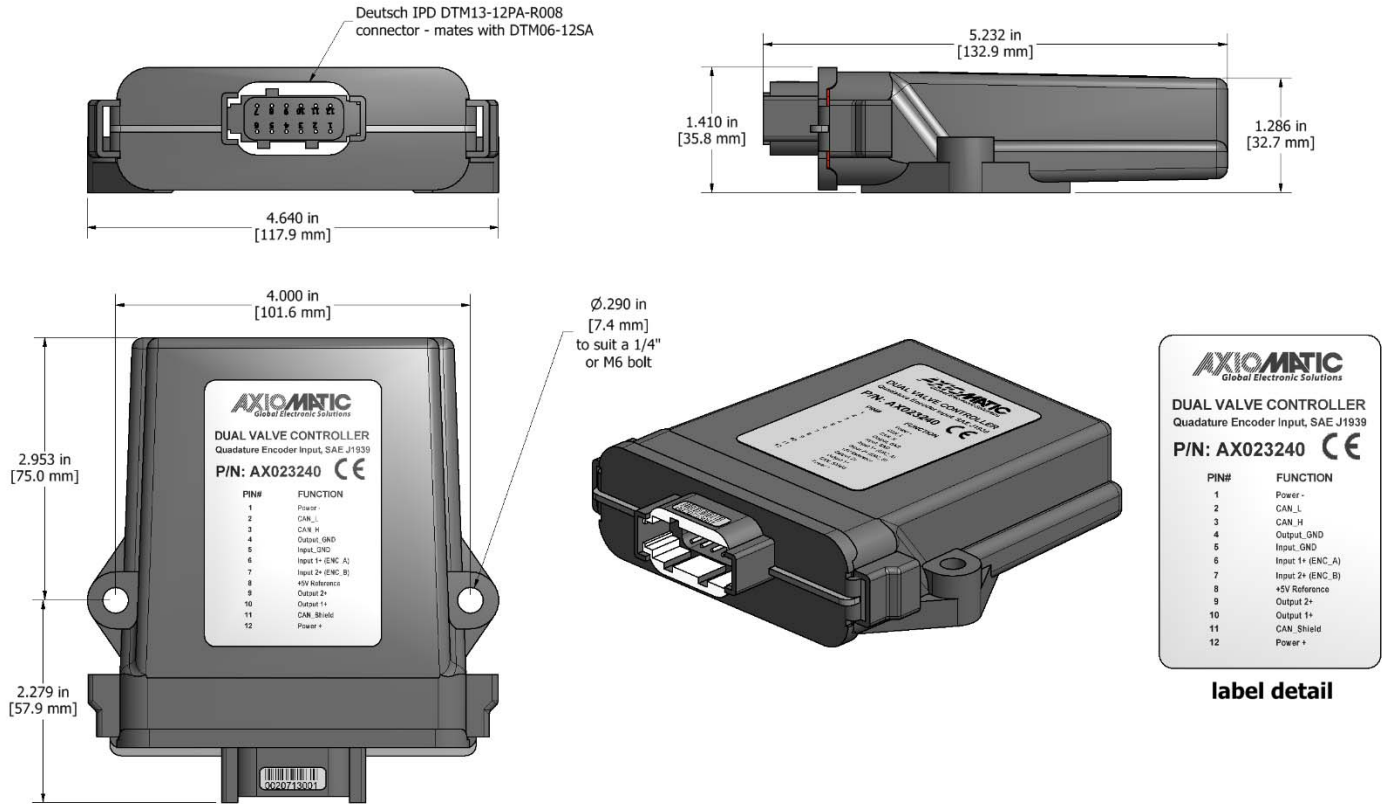


Figure 9: Housing Dimensions

Table 23: Connector Pinout

CAN and I/O Connector	
Pin #	Description (Notes)
1	BATT -
2	CAN_L
3	CAN_H
4	P_GND (Out 1 and Out 2)
5	Analog_GND (Input 1 and Input 2)
6	Input 1+
7	Input 2+
8	+5V Ref
9	Output 2+ (Default: Not Used)
10	Output 1+
11	CAN_Shield
12	BATT +

## 2.2. Mounting Instructions

### NOTES & WARNINGS

- Do not install near high-voltage or high-current devices.
- Note the operating temperature range. All field wiring must be suitable for that temperature range.
- Install the unit with appropriate space available for servicing and for adequate wire harness access (15 cm) and strain relief (30 cm).
- Do not connect or disconnect the unit while the circuit is live, unless the area is known to be non-hazardous.

### MOUNTING

The module is designed for mounting on the valve block. If it is mounted without an enclosure, the controller should be mounted horizontally with connectors facing left or right, or with the connectors facing down, to reduce likelihood of moisture entry.

Mask all labels if the unit is to be repainted, so label information remains visible.

Mounting legs include holes sized for ¼” bolts. The bolt length will be determined by the end-user’s mounting plate thickness. Typically 20 mm (3/4 inch) is adequate.

If the module is mounted away from the valve block, no wire or cable in the harness should exceed 30 meters in length. The power input wiring should be limited to 10 meters.

### CONNECTIONS

Use the following Deutsch IPD mating plugs to connect to the integral receptacles. Wiring to these mating plugs must be in accordance with all applicable local codes. Suitable field wiring for the rated voltage and current must be used. The rating of the connecting cables must be at least 85°C. For ambient temperatures below –10°C and above +70°C, use field wiring suitable for both minimum and maximum ambient temperature.

Refer to the respective Deutsch IPD datasheets for usable insulation diameter ranges and other instructions.

Receptacle Contacts	Mating Sockets as appropriate (Refer to <a href="http://www.laddinc.com">www.laddinc.com</a> for more information on the contacts available for this mating plug.)
Mating Connector	DTM06-12SA, DTM06-12SB, 2 wedges WM12S, 24 contacts (0462-201-20141)

### 3. OVERVIEW OF J1939 FEATURES

---

The software was designed to provide flexibility to the user with respect to messages sent to and from the ECU by providing:

- Configurable ECU Instance in the NAME (to allow multiple ECUs on the same network)
- Configurable Transmit PGN and SPN Parameters
- Configurable Receive PGN and SPN Parameters
- Sending DM1 Diagnostic Message Parameters
- Reading and reacting to DM1 messages sent by other ECUs
- Diagnostic Log, maintained in non-volatile memory, for sending DM2 messages

#### 3.1. Introduction To Supported Messages

The ECU is compliant with the standard SAE J1939, and supports the following PGNs

##### From J1939-21 - Data Link Layer

- Request 59904 (\$00EA00)
- Acknowledgment 59392 (\$00E800)
- Transport Protocol – Connection Management 60416 (\$00EC00)
- Transport Protocol – Data Transfer Message 60160 (\$00EB00)
- PropB Transmit, Default Measured Inputs Feedback Message 65280 (\$00FF00)
- PropB Transmit, Default Proportional Outputs Target Message 65296 (\$00FF10)
- PropB Transmit, Default Proportional Outputs Feedback Message 65312 (\$00FF20)
- PropB Transmit, Default Digital I/O State Feedback Message 65328 (\$00FF30)
- PropB Receive, Default Output Control Data Message 65408 (\$00FF80)
- PropB Receive, Default Output Enable Data Message 65424 (\$00FF90)
- PropB Receive, Default Output Override Data Message 65440 (\$00FFA0)
- PropB Receive, Default PID Feedback Data Message 65456 (\$00FFB0)

Note: Any Proprietary B PGN in the range 65280 to 65535 (\$00FF00 to \$00FFFF) can be selected

Note: The Proprietary A PGN 61184 (\$00EF00) can also be selected for any of the messages

##### From J1939-73 - Diagnostics

- DM1 – Active Diagnostic Trouble Codes 65226 (\$00FECA)
- DM2 – Previously Active Diagnostic Trouble Codes 65227 (\$00FECB)
- DM3 – Diagnostic Data Clear/Reset for Previously Active DTCs 65228 (\$00FECC)
- DM11 - Diagnostic Data Clear/Reset for Active DTCs 65235 (\$00FED3)
- DM14 – Memory Access Request 55552 (\$00D900)
- DM15 – Memory Access Response 55296 (\$00D800)
- DM16 – Binary Data Transfer 55040 (\$00D700)

##### From J1939-81 - Network Management

- Address Claimed/Cannot Claim 60928 (\$00EE00)
- Commanded Address 65240 (\$00FED8)

##### From J1939-71 – Vehicle Application Layer

- Software Identification 65242 (\$00FEDA)

None of the application layer PGNs are supported as part of the default configurations, but they can be selected as desired for either transmit or received function blocks.

Setpoints are accessed using standard Memory Access Protocol (MAP) with proprietary addresses. The Electronic Assistant <sup>®</sup> EA (EA) allows for quick and easy configuration of the unit over the CAN network.

### 3.2. NAME, Address and Software ID

#### J1939 NAME

The High Temperature 2 Input 2 Output Valve controller ECU has the following defaults for the J1939 NAME. The user should refer to the SAE J1939/81 standard for more information on these parameters and their ranges.

Arbitrary Address Capable	Yes
Industry Group	0, Global
Vehicle System Instance	0
Vehicle System	0, Non-specific system
Function	125, I/O Controller (Axiomatic-specific)
Function Instance	24, Axiomatic AX023240, Quadrature Encoder Input 2 Output High Temperature Controller
<b>ECU Instance</b>	<b>0, First Instance</b>
Manufacture Code	162, Axiomatic Technologies Corporation
Identity Number	Variable, uniquely assigned during factory programming for each ECU

The ECU Instance is a configurable setpoint associated with the NAME. Changing this value will allow multiple ECUs of this type to be distinguishable by other ECUs (including the Electronic Assistant <sup>®</sup>) when they are all connected on the same network.

#### ECU Address

The default value of this setpoint is 128 (0x80), which is the preferred starting address for self-configurable ECUs as set by the SAE in J1939 tables B3 to B7. The EA will allow the selection of any address between 0 to 253, and ***it is the user's responsibility to select an address that complies with the standard.*** The user must also be aware that since the unit is arbitrary address capable, if another ECU with a higher priority NAME contends for the selected address, the High Temperature Quadrature Encoder Input 2 Output Valve controller will continue select the next highest address until it find one that it can claim. See J1939/81 for more details about address claiming.

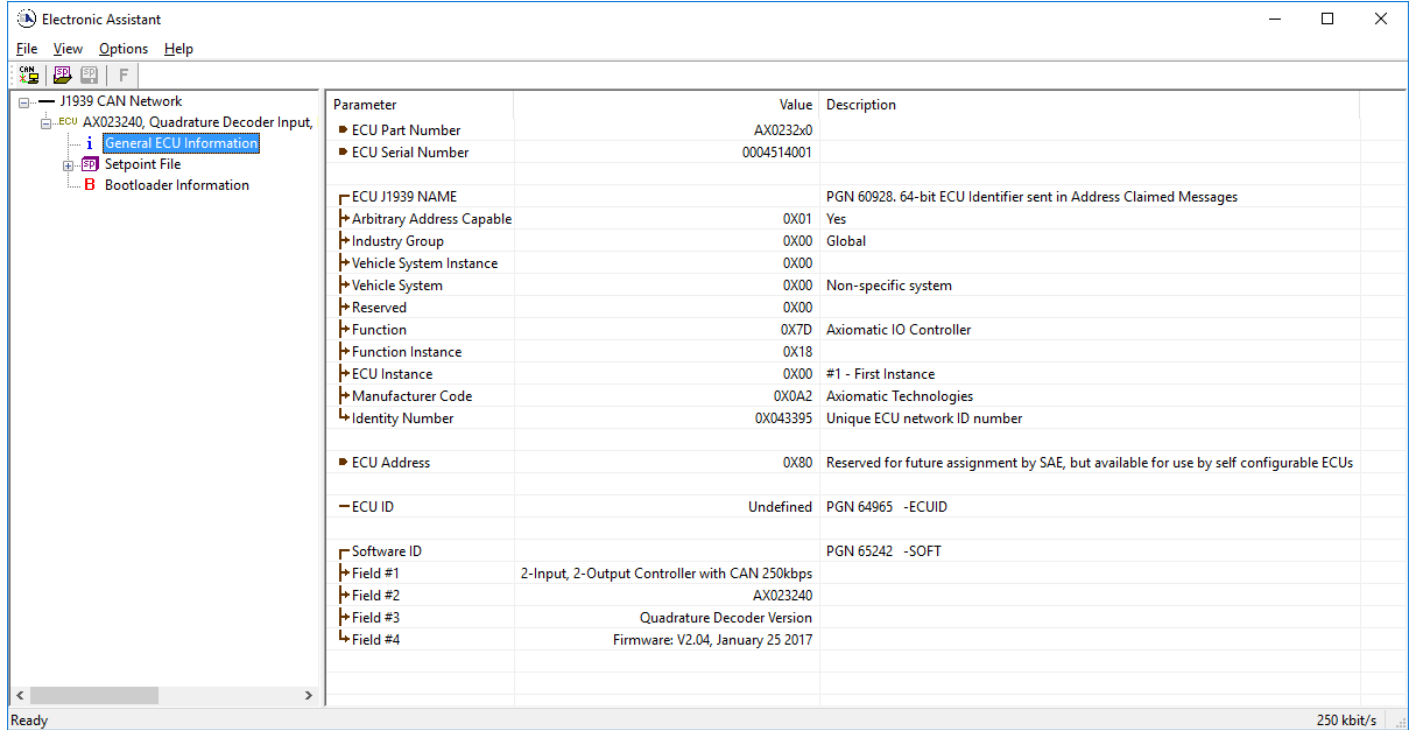
#### Software Identifier

PGN 65242	Software Identification	- SOFT	
Transmission Repetition Rate:	On request		
Data Length:	Variable		
Extended Data Page:	0		
Data Page:	0		
PDU Format:	254		
PDU Specific:	218 PGN Supporting Information:		
Default Priority:	6		
Parameter Group Number:	65242 (0xFEDA)		
Start Position	Length	Parameter Name	SPN
1	1 Byte	Number of software identification fields	965
2-n	Variable	Software identification(s), Delimiter (ASCII “*”)	234

For the High Temperature Quadrature Encoder Input 2 Output Valve controller ECU, Byte 1 is set to 5, and the identification fields are as follows

**(Part Number)\*(Version)\*(Date)\*(Owner)\*(Description)**

EA shows all this information in “General ECU Information”, as shown below

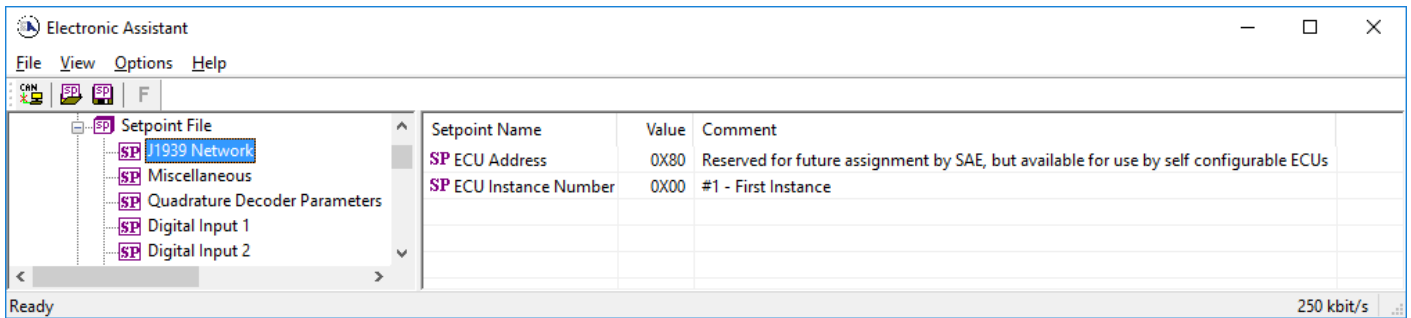


*Note: The information provided in the Software ID is available for any J1939 service tool which supports the PGN -SOFT.*

## 4. ECU SETPOINTS ACCESSED WITH ELECTRONIC ASSISTANT

Many setpoints have been reference throughout this manual. This section describes in detail each setpoint, and their defaults and ranges. For more information on how each setpoint is used by the High Temperature Quadrature Encoder Input 2 Output, refer to the relevant section of the User Manual.

### 4.1. J1939 Network Setpoints



Name	Range	Default	Notes
ECU Address	0 to 253	128 (0x80)	Preferred address for a self-configurable ECU
ECU Instance Number	Drop List	0, #1 – First Instance	Per J1939-81

If non-default values for the “**ECU Instance Number**” or “**ECU Address**” are used, they will not be updated during a setpoint file flash. These parameters need to be changed manually in order to prevent other units on the network to be affected. When they are changed, the controller will claim its new address on the network. It is recommended to close and re-open the CAN connection on EA after the file is loaded so that only the new NAME and address are showing in the J1939 CAN Network ECU list.

### 4.2. Miscellaneous Setpoints

The Miscellaneous setpoints primarily deal setpoints which apply to the overall controller and/or a set of setpoint groups. Refer to the notes for more information about each setpoint.

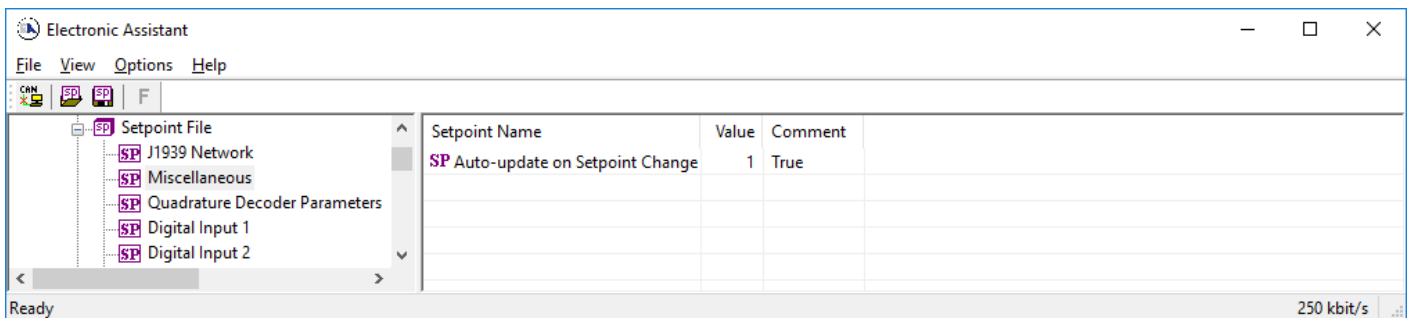


Figure 10: Screen capture of Default Miscellaneous Setpoints:

Name	Range	Default	Notes
Auto-update on Setpoint Change	True/False	True	By default this setpoint is set to <i>TRUE</i>

and it refers to how the controller will update setpoint groups in accordance to a setpoint change. When this setpoint is *FALSE* other setpoint groups in the controller that use the setpoint will not get affected.

### 4.3. Quadrature Encoder Parameter Setpoints

The Quadrature Encoder unit parameters can be configured to the controller using these setpoints. The configuration options include scaler for step count, polarity of the direction signal, step count offset and how many pulses the Encoder generates per revolution.

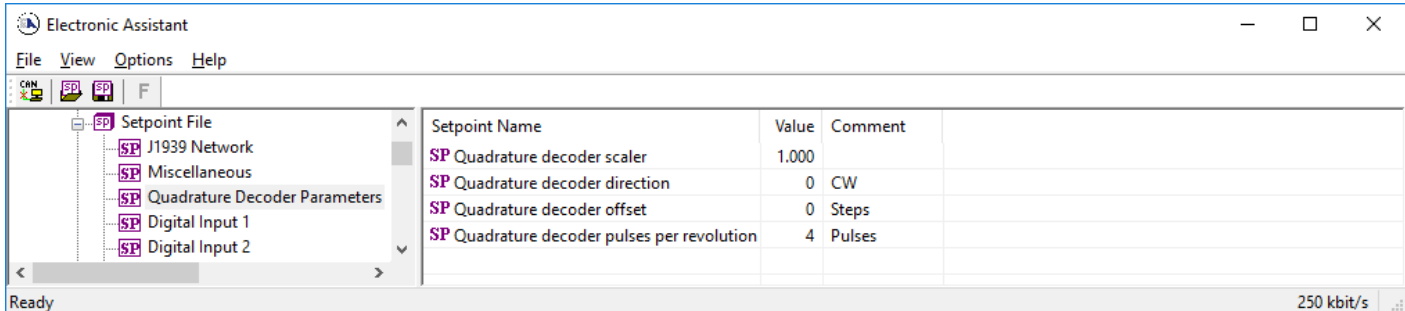


Figure 11: Screen capture of Default Quadrature Encoder Parameter Setpoints

### 4.4. Quadrature Encoder / Digital Input Setpoints

The Input Function block is defined in Section 1.2. Please refer to that section for detailed information about how these setpoints are used.

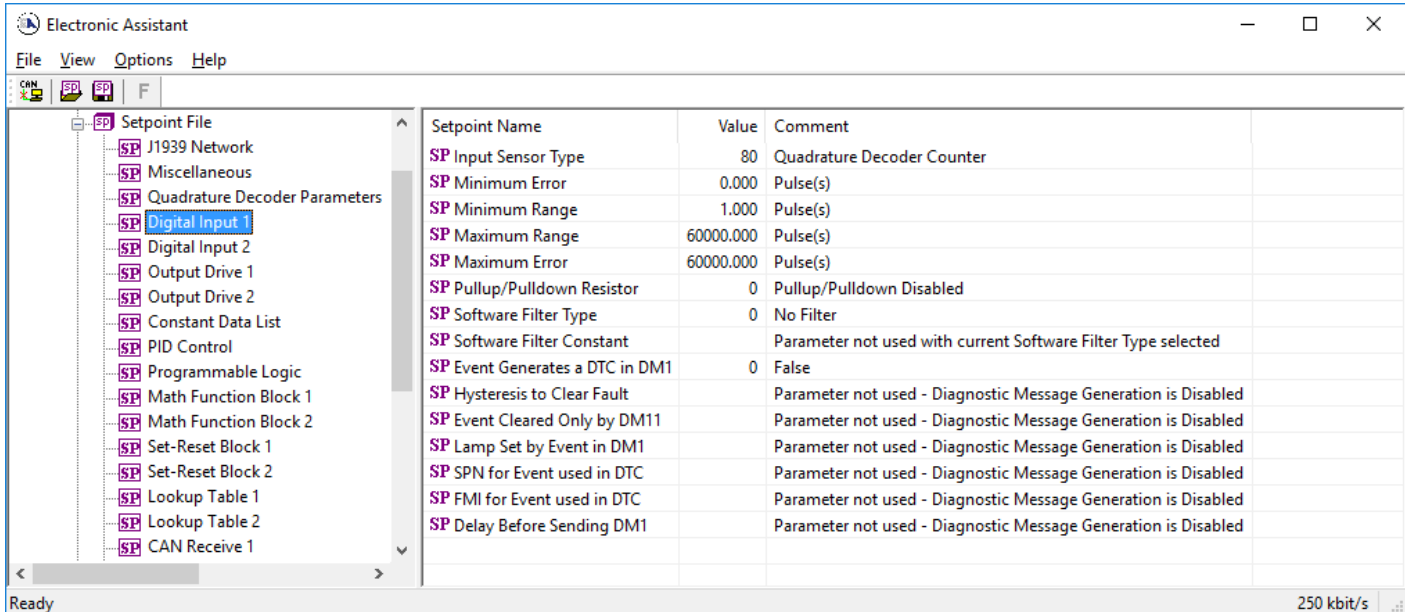


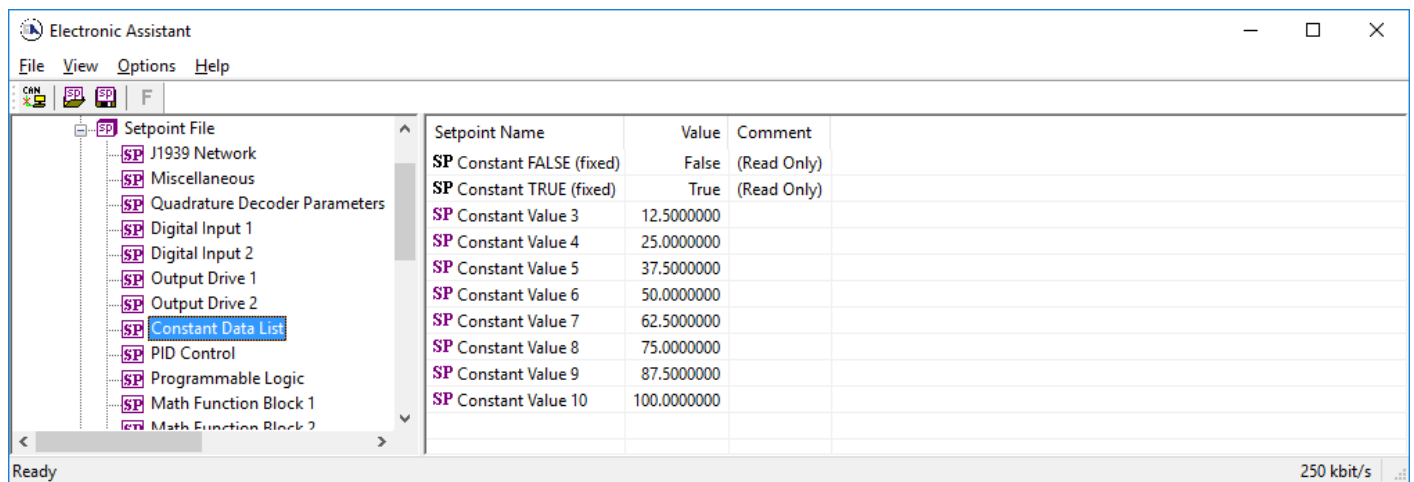
Figure 12: Screen capture of Default Digital Input 1 Setpoints

## 4.5. Constant Data List Setpoints

The Constant Data List function block is provided to allow the user to select values as desired for various logic block functions. Throughout this manual, various references have been made to constants, as summarized in the examples listed below.

- a) Programmable Logic: Constant “**Table X = Condition Y, Argument 2**”, where X and Y = 1 to 3
- b) Math Function: Constant “**Math Input X**”, where X = 1 to 2

The first two constants are fixed values of 0 (False) and 1 (True) for use in binary logic. The remaining 8 constants are fully user programmable to any value between +/- 1,000,000. The default values (shown below) are arbitrary and should be configured by the user as appropriate for their application.



Setpoint Name	Value	Comment
SP Constant FALSE (fixed)	False	(Read Only)
SP Constant TRUE (fixed)	True	(Read Only)
SP Constant Value 3	12.50000000	
SP Constant Value 4	25.00000000	
SP Constant Value 5	37.50000000	
SP Constant Value 6	50.00000000	
SP Constant Value 7	62.50000000	
SP Constant Value 8	75.00000000	
SP Constant Value 9	87.50000000	
SP Constant Value 10	100.00000000	

Figure 13: Screen capture of Default Constant Data List Setpoints

## 4.6. Lookup Table Setpoints

The Lookup Table function block is defined in Section 1.5. Please refer there for detailed information about how all these setpoints are used. As this function block’s X-Axis defaults are defined by the **X-Axis Source** selected from Table 4, there is nothing further to define in terms of defaults and ranges beyond that which is described in Section 1.5. Recall, the X-Axis values will be automatically updated if the min/max range of the selected source is changed if **Auto-update on Setpoint Change** setpoint is *TRUE*. Otherwise, there will be no changes to the X-Axis values and is the user’s responsibility to ensure the values are appropriately selected.

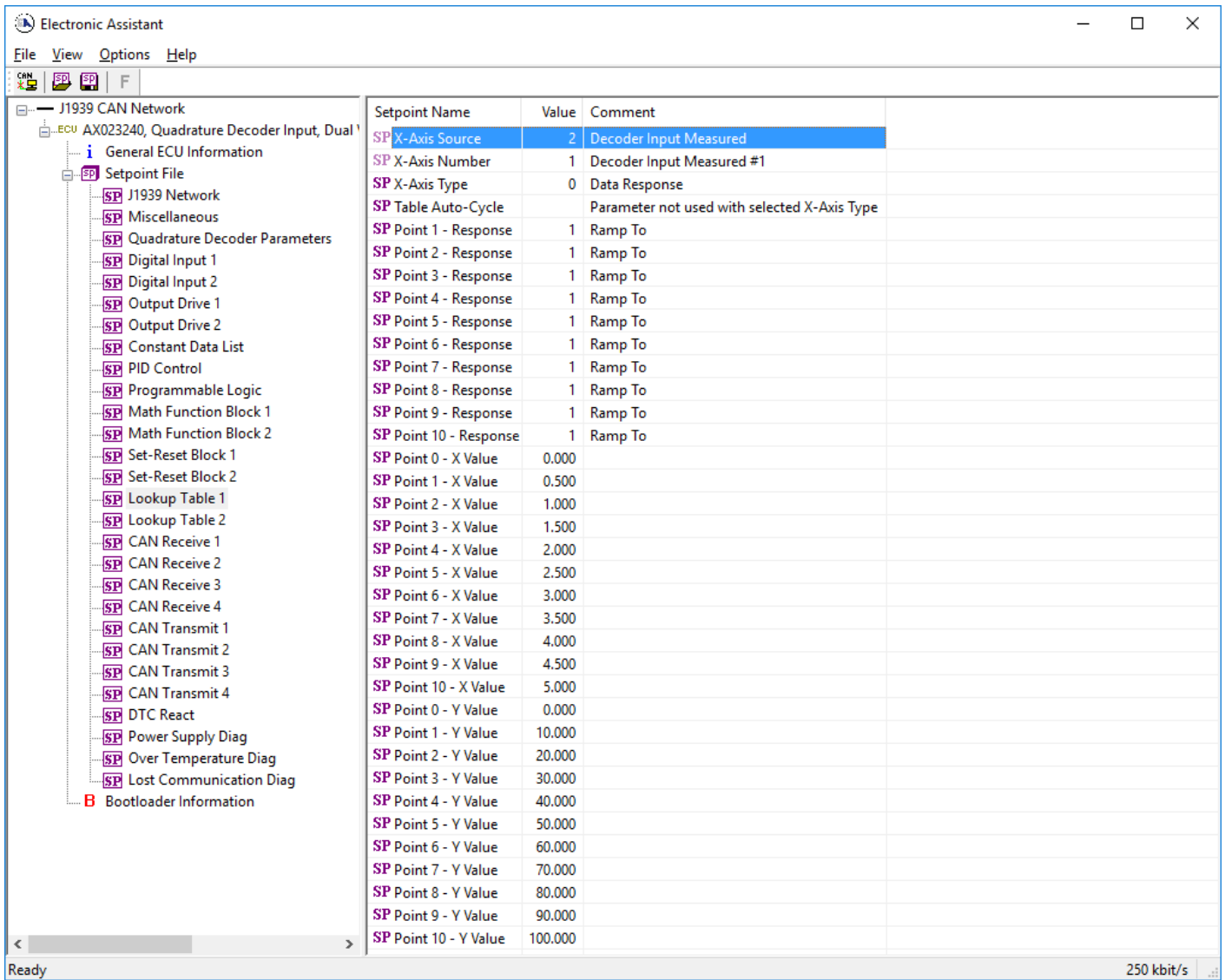


Figure 14: Screen capture of Default Lookup Table 1 Setpoints

Note: In the screen capture shown above, the “X-Axis Source” has been changed from its default value in order to enable the function block.

#### 4.7. Programmable Logic Setpoints

The Programmable Logic function block is defined in Section 1.7. Please refer there for detailed information about how all these setpoints are used.

As this function block is disabled by default, there is nothing further to define in terms of defaults and ranges beyond that which is described in Section 1.7. The screen capture below shows how the setpoints referenced in that section appear on EA.

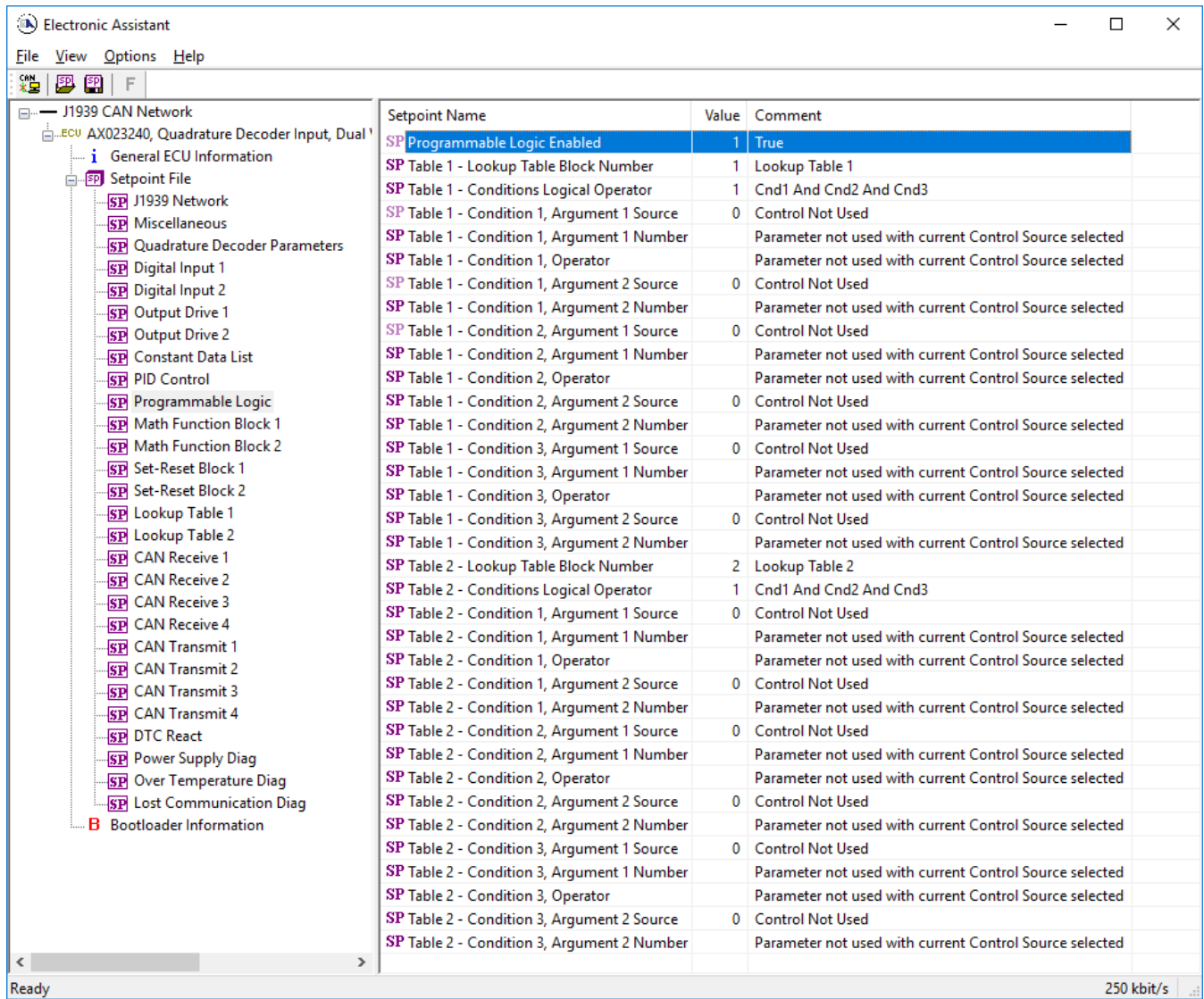


Figure 15: Screen capture of Default Programmable Logic Setpoints

*Note: In the screen capture shown above, the “Programmable Logic Block Enabled” has been changed from its default value in order to enable the function block.*

*Note: The default values for the Argument1, Argument 2 and Operator are all the same across all the Programmable Logic function blocks, and must therefore be changed by the user as appropriate before this can be used.*

## 4.8. Math Function Setpoints

The Math Function block is defined in Section 1.6. Please refer there for detailed information about how all these setpoints are used.

Setpoint Name	Value	Comment
SP Math Function Enabled	1	True
SP Function 1 Input A Source	0	Control Not Used
SP Function 1 Input A Number		Parameter not used with current Control Source selected
SP Function 1 Input A Minimum		Parameter not used with current Control Source selected
SP Function 1 Input A Maximum		Parameter not used with current Control Source selected
SP Function 1 Input A Scaler		Parameter not used with current Control Source selected
SP Function 1 Input B Source	0	Control Not Used
SP Function 1 Input B Number		Parameter not used with current Control Source selected
SP Function 1 Input B Minimum		Parameter not used with current Control Source selected
SP Function 1 Input B Maximum		Parameter not used with current Control Source selected
SP Function 1 Input B Scaler		Parameter not used with current Control Source selected
SP Math Function 1 Operation		Parameter not used with current Control Source selected
SP Function 2 Input B Source	0	Control Not Used
SP Function 2 Input B Number		Parameter not used with current Control Source selected
SP Function 2 Input B Minimum		Parameter not used with current Control Source selected
SP Function 2 Input B Maximum		Parameter not used with current Control Source selected
SP Function 2 Input B Scaler		Parameter not used with current Control Source selected
SP Math Function 2 Operation (Input A = Result of Function 1)		Parameter not used with current Control Source selected
SP Function 3 Input B Source	0	Control Not Used
SP Function 3 Input B Number		Parameter not used with current Control Source selected
SP Function 3 Input B Minimum		Parameter not used with current Control Source selected
SP Function 3 Input B Maximum		Parameter not used with current Control Source selected
SP Function 3 Input B Scaler		Parameter not used with current Control Source selected
SP Math Function 3 Operation (Input A = Result of Function 2)		Parameter not used with current Control Source selected
SP Math Output Minimum Range	0.00	
SP Math Output Maximum Range	100.00	

Figure 16: Screen capture of Default Math Function 1 Setpoints

Note: In the screen capture shown above, the “Math Function Enabled” has been changed from its default value in order to enable the function block

## 4.9. Set-Reset Function Block Setpoints

The Set-Reset function block is defined in Section 1.9. Please refer there for detailed information about how all these setpoints are used.

Setpoint Name	Value	Comment
SP Block Enabled	1	True
SP Reset Source	0	Control Not Used
SP Reset Number		Parameter not used with current Control Source selected
SP Set Source	0	Control Not Used
SP Set Number		Parameter not used with current Control Source selected

Figure 17: Screen capture of Default Set-Reset Block 1 Setpoints

Note: In the screen capture shown above, the “Block Enabled” has been changed from its default value in order to enable the function block

### 4.10. CAN Transmit Setpoints

The CAN Transmit function block is defined in Section 0. Please refer there for detailed information about how all these setpoints are used.

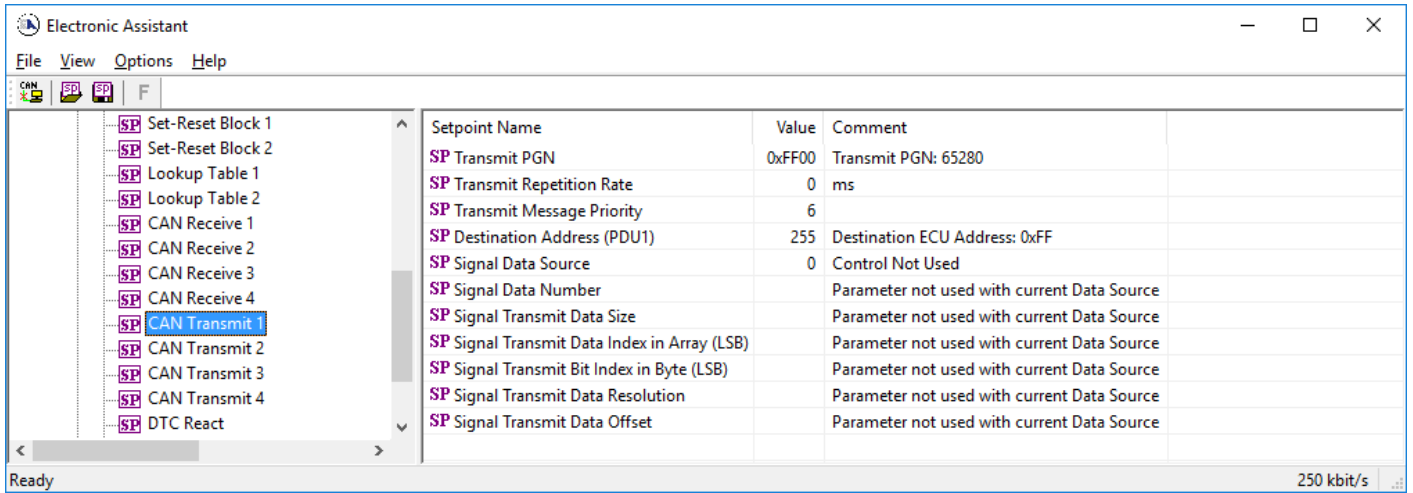


Figure 18: Screen capture of Default CAN Transmit 1 Setpoints

### 4.11. CAN Receive Setpoints

The CAN Receive function block is defined in Section 1.12. Please refer there for detailed information about how all these setpoints are used.

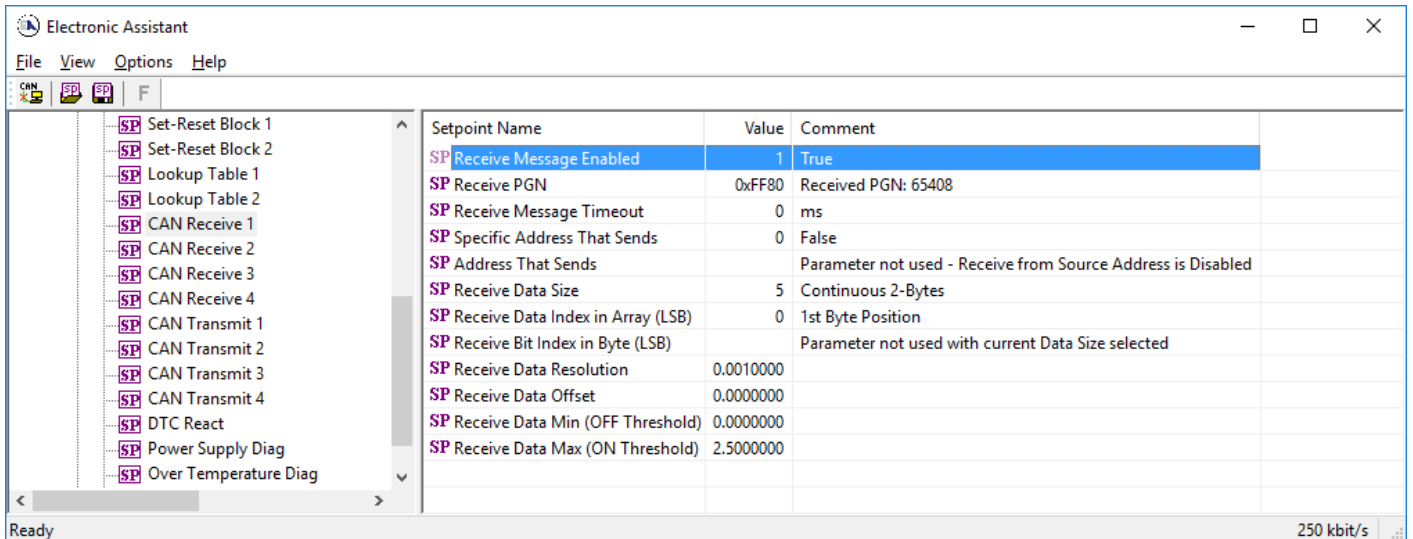


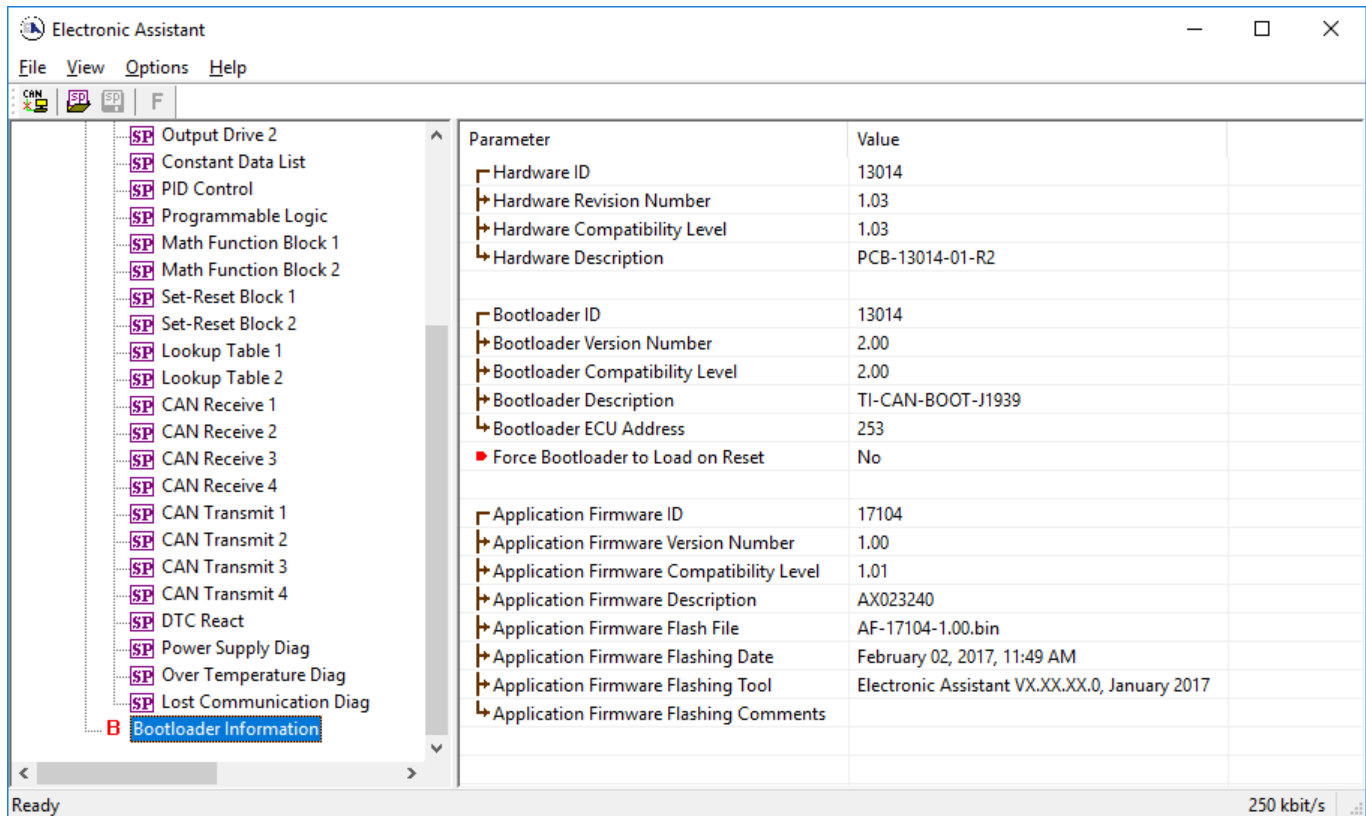
Figure 19: Screen capture of Default CAN Receive 1 Setpoints

Note: In the screen capture shown above, the “Receive Message Enabled” has been changed from its default value in order to enable the function block

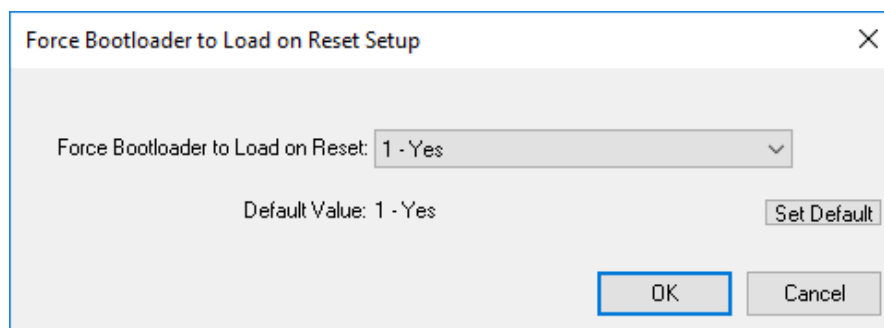
## 5. REFLASHING OVER CAN WITH EA® BOOTLOADER

The AX023240 can be upgraded with new application firmware using the **Bootloader Information** section. This section details the simple step-by-step instructions to upload new firmware provided by Axiomatic onto the unit via CAN, without requiring it to be disconnected from the J1939 network.

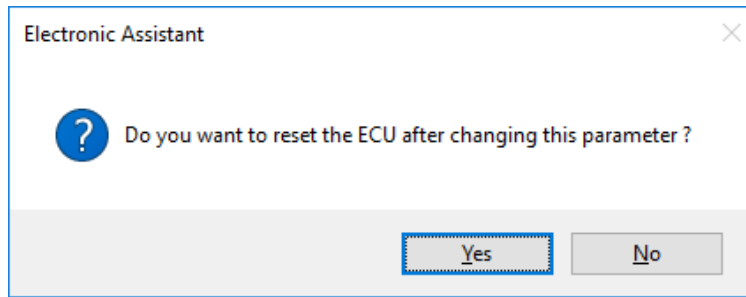
1. When EA first connects to the ECU, the **Bootloader Information** section will display the following information.



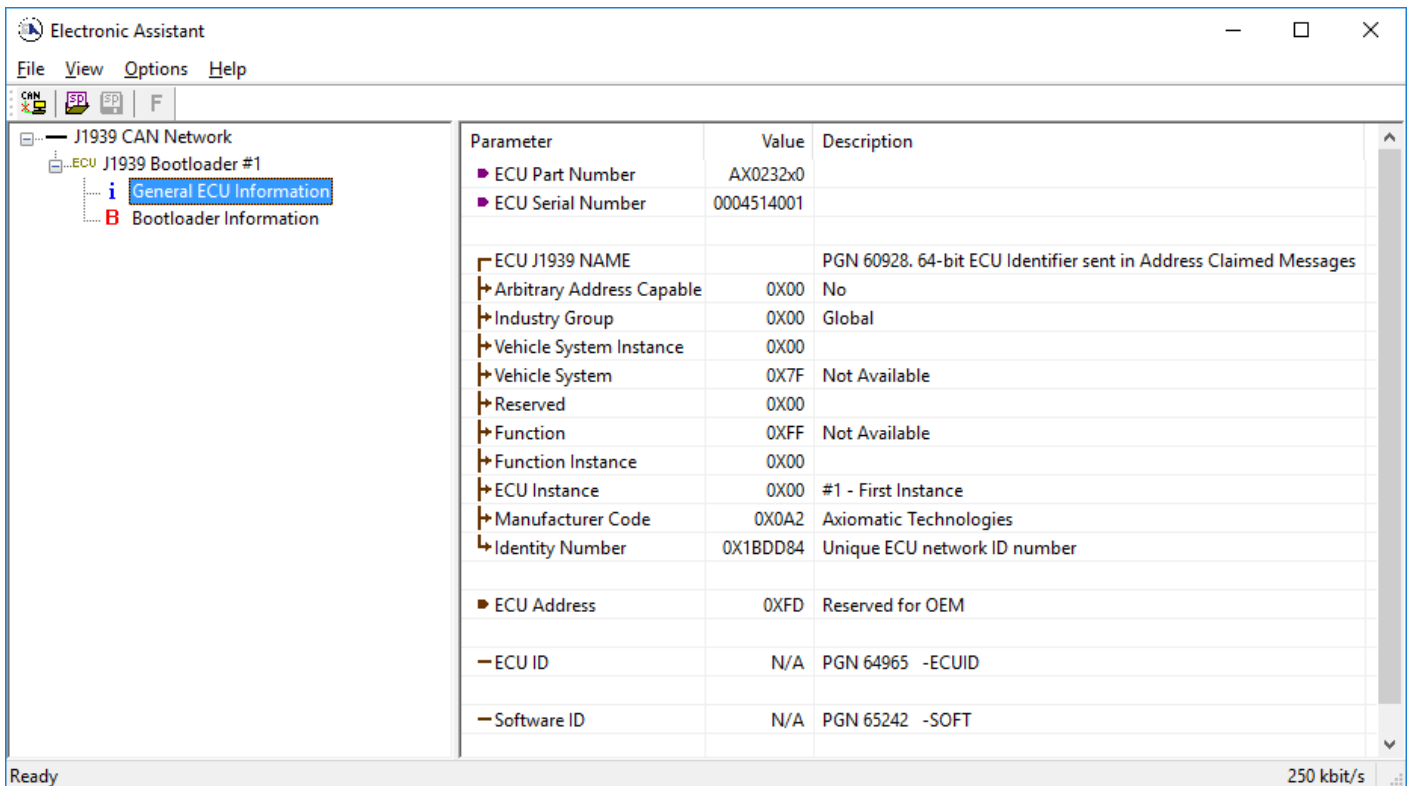
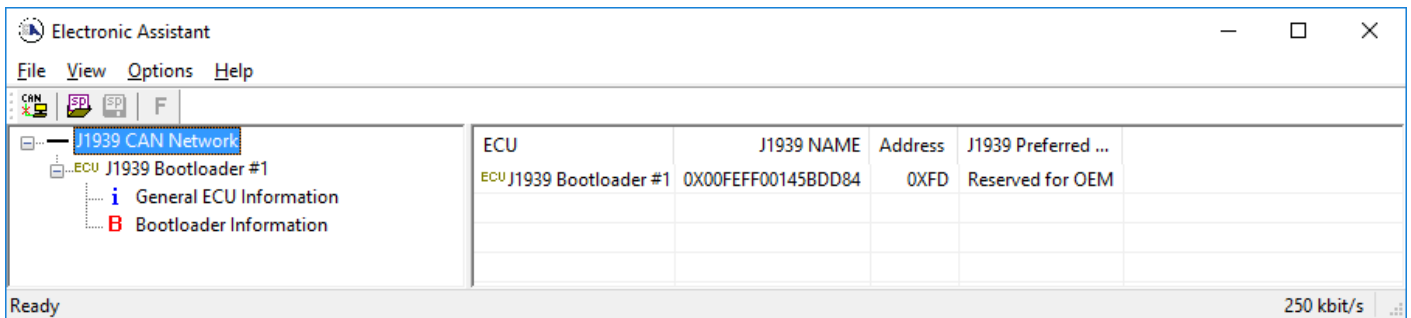
2. To use the bootloader to upgrade the firmware running on the ECU, change the variable **“Force Bootloader To Load on Reset”** to Yes.



3. When the prompt box asks if you want to reset the ECU, select Yes.

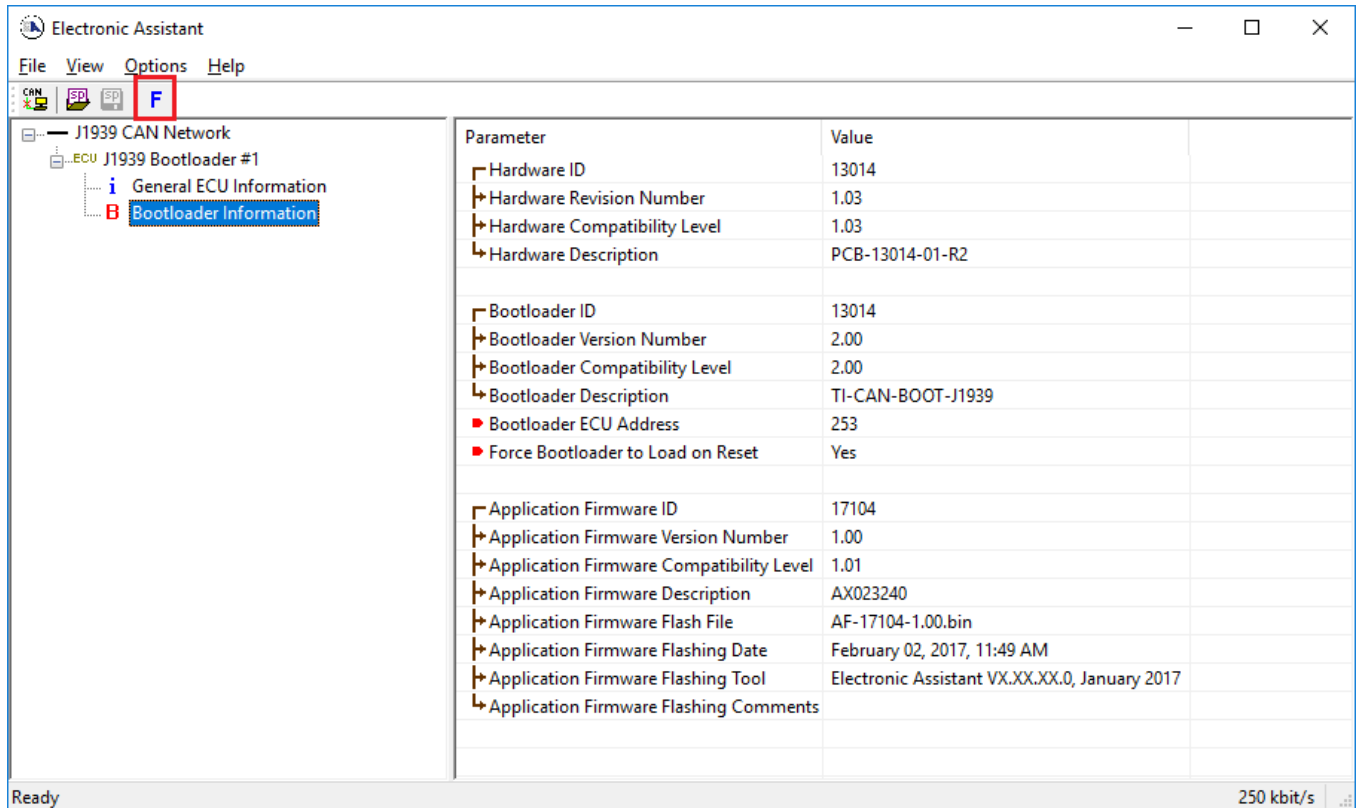


- Upon reset, the ECU will no longer show up on the J1939 network as an AX023240 but rather as **J1939 Bootloader #1**.



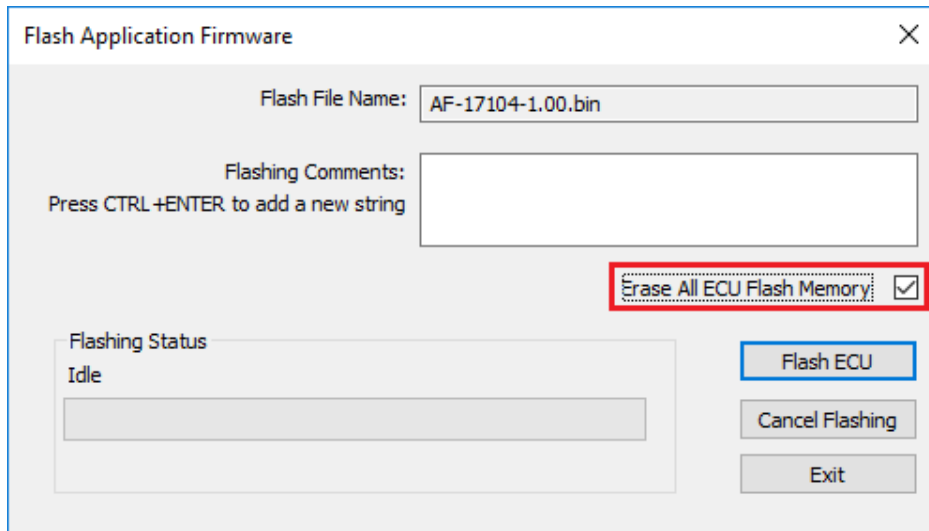
*Note that the bootloader is NOT Arbitrary Address Capable. This means that if you want to have multiple bootloaders running simultaneously (not recommended) you would have to manually change the address for each one before activating the next, or there will be address conflicts, and only one ECU would show up as the bootloader. Once the 'active' bootloader returns to regular functionality, the other ECU(s) would have to be power cycled to re-activate the bootloader feature.*


- When the **Bootloader Information** section is selected, the same information is shown as when it was running the AX023240 firmware, but in this case the **F**lashing feature has been enabled.



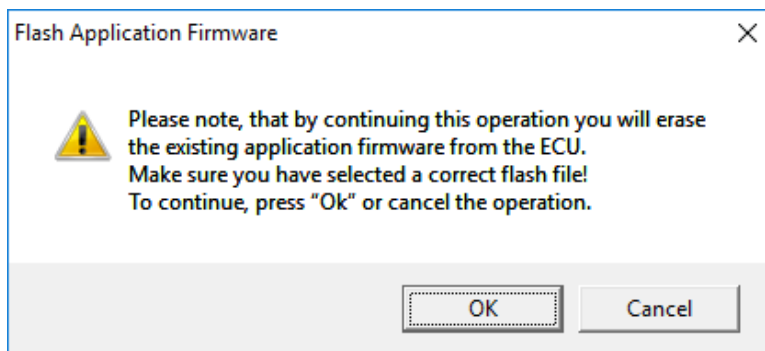
- Select the **F**lashing button and navigate to where you had saved the **AF-17104-x.yy.bin** file sent from Axiomatic. (Note: only binary (.bin) files can be flashed using the EA tool)
- Once the Flash Application Firmware window opens, you can enter comments such as "Firmware upgraded by [Name]" if you so desire. This is not required, and you can leave the field blank if you do not want to use it.

Note: You do not have to date/time-stamp the file, as this is done automatically by the EA tool when you upload the new firmware.

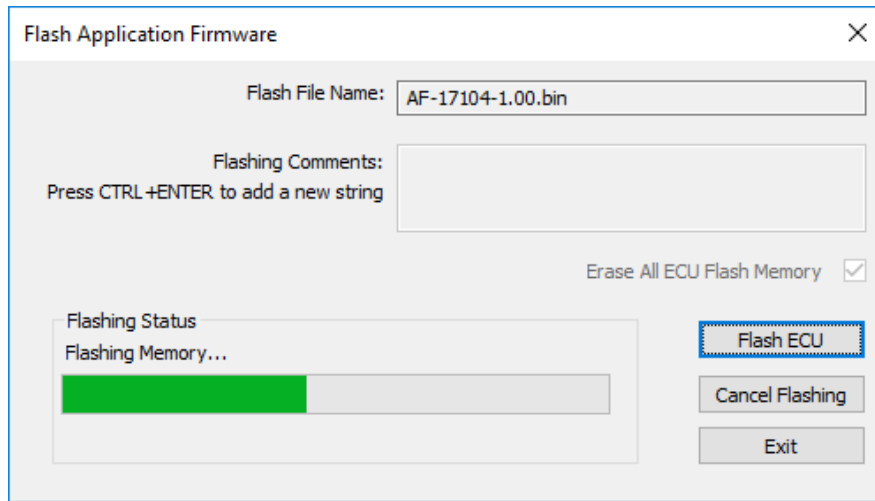


 **NOTE:** Please make sure that the “Erase All ECU Flash Memory” box is checked. Selecting this will erase ALL data stored in non-volatile flash. It will also erase any configuration of the setpoints that might have been done to the ECU and reset all setpoints to their factory defaults. **If this box is left unchecked, the firmware upload will FAIL.**

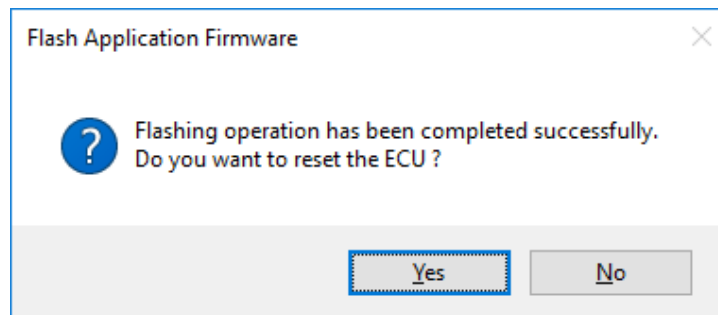
When proceeding to flashing with the “Erase All ECU Flash Memory” option checked, EA will warn the user about erasing all setpoint data.



8. A progress bar will show how much of the firmware has been sent as the upload progresses. The more traffic there is on the J1939 network, the longer the upload process will take.



9. Once the firmware has finished uploading, a message will popup indicating the successful operation. If you select to reset the ECU, the new version of the AX023240 application will start running, and the ECU will be identified as such by EA. Otherwise, The next time the ECU is power-cycled, the AX023240 application will run rather than the bootloader function.



Note: If at any time during the upload the process is interrupted, the data is corrupted (bad checksum) or for any other reason the new firmware is not correct, i.e. bootloader detects that the file loaded was not designed to run on the hardware platform, the bad or corrupted application will not run. Rather, when the ECU is reset or power-cycled the **J1939 Bootloader** will continue to be the default application until valid firmware has been successfully uploaded into the unit.

## 6. TECHNICAL SPECIFICATIONS

### 6.1. Power Supply, CAN and Reference Voltage

Power Supply Input - Nominal	12 or 24Vdc nominal (9...60 Vdc power supply range)
Protection	Reverse polarity protection is provided. Surge protection up to 65V is provided. Overvoltage shutdown of the output load is provided. Undervoltage protection (hardware and software shutdown at 7.5V) is provided.
CAN	SAE J1939 Commands 500 kbps and 1 Mbps baud rate models are available. See Ordering part numbers.
Voltage Reference	One provided 5V +/- 0.2% error Can source up to 50mA without derating
Analog GND Reference	One provided

### 6.2. Inputs

Universal Signal Inputs	2 fully independent digital inputs are provided. Refer to Table 1.0 All inputs are user selectable as Quadrature Encoder or Digital input types. Inputs are sampled multiple times per millisecond. Protected against shorts to GND or +Vps (up to 60 Vdc) All input channels can handle negative voltage inputs down to -2VDC due to voltage spikes or noise. Response time to change at the input 2 mSec +/- 1 mSec (without software filtering) unless otherwise noted.
-------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Table 1.0 – Input – User Selectable Options**

Digital Input Functions	Discrete Input, Quadrature Encoder Input (Steps, Direction, Speed in RPM)
Digital Input Level	12V
Quadrature Encoder Input	Configurable 1kΩ pullup or 10kΩ pulldown resistor (to GND) resistor which can also be disabled (floating input) Rising/Falling edge threshold 2.0V +/- 0.1V Number or pulses per revolution configurable. Custom scaler for Encoder steps counter value. Configurable polarity of direction of rotation.
Digital Input	Normal, Inverse or Latched (pushbutton) Configurable 1kΩ pullup or 10kΩ pulldown resistor (to GND) which can also be disabled (floating input) Rising/Falling edge threshold 2.0V +/- 0.1V Input debouncing time selectable

### 6.3. Outputs

CAN	SAE J1939 Messages
Universal Outputs	<p>Two independent software controlled outputs selectable as: Proportional Current; Hotshot Digital; PWM Duty Cycle; Proportional Voltage; or On/Off Digital types</p> <p>Half-bridge outputs, current sensing, grounded load. High side sourcing up to 3A</p> <p>All output types have configurable minimum and maximum output levels within the range for the type selected.</p> <p>Current Outputs: 1mA resolution, accuracy +/- 2% error Software controlled PID current Range 0 to 3000 mA Fully configurable dither superimposed on top of output current Configurable from 50 to 400Hz amplitude High frequency output drive at 25kHz</p> <p>Voltage Outputs: 0.1V resolution, accuracy +/- 3% error Average voltage output based on unit power supply High frequency drive at 25kHz Additional external filtering is required to create a DC voltage</p> <p>PWM Outputs: 0.1% resolution, accuracy +/- 1% error Range 0 to 100% Output Frequency: 1 Hz to 25 kHz Configurable frequency ONLY if no current output types are used, otherwise default 25kHz is used</p> <p>Digital On/Off: Load at supply voltage must not draw more than 3A.</p>
Response Time	Contact Axiomatic.
Protection	Fully protected against short circuit to ground or +Vps Grounded short circuit protection will engage at 4.5A +/- 0.5A. Unit will fail safe in the case of a short-circuit condition, and is self-recovering when the short is removed.
Power GND Reference	One Provided

### 6.4. General Specifications

Quiescent Current	109 mA @ 12Vdc Typical; 66 mA @ 24Vdc Typical
Microprocessor	TI TMS320F2806x, 32-bit, 256 KB flash program memory, 100 KB RAM
EMC Compliance	CE marking
Vibration	Random Vibration: 7.7 Grms peak Sinusoidal Component: 10 g peak Based on MIL-STD-202G, Methods 204G and 214A
Diagnostics	Each input and output channel can be configured to send diagnostic messages to the J1939 CAN network if the I/O goes out of range. Diagnostic data is stored in a non-volatile log. Refer to the User Manual for details.
Additional Fault Feedback	There are several types of faults that the controller will detect and provide a response: unit power supply undervoltage and overvoltage, microprocessor over temperature and lost communication. They can be sent to the J1939 CAN bus.
Control Logic	User configurable functionality using Electronic Assistant® service tool
Communications	Compliant to SAE CAN J1939 Standard 1 CAN port (SAE J1939) CANopen® is available on request.
CAN User Interface	Compliant to SAE CAN J1939 Standard  Interfaces with the Electronic Assistant®, P/N: AX070502, for Windows operating systems. It comes with a royalty-free license for use. To use the Electronic Assistant, an USB-CAN converter links the device's CAN port to a Windows-based PC.
CAN Response Time	Per the J1939 standard, the maximum recommended transmit rate for any message is 10ms. Response time of feedback on the CAN to changes at the I/O will be a combination of the I/O type's response time and the configurable software filtering, ramps, delays, etc. that were selected in the application.
Reflashing Software over CAN	Reflash software over the CAN bus using the Electronic Assistant®.
Packaging	High Temperature Nylon housing - Deutsch IPD PCB Enclosure (EEC-325X4B) 4.62 x 5.24 x 1.43 inches 117.42 x 133.09 x 36.36 mm (W x L x H excluding mating plugs) Refer to the dimensional drawing.

Protection	IP67 rating for the product assembly
Weight	0.50 lb. (0.23 kg)
Temperature Rating	-40°C to +125°C (-40°F to 257°F)

Electrical Connections	<p>12 pin Deutsch IPD connector P/N: DTM13-12PA-R008 20 AWG wire is recommended for use with contacts 0462-201-20141.</p> <table border="1"> <thead> <tr> <th colspan="2">CAN and I/O Connector</th> </tr> <tr> <th>Pin #</th> <th>Description (Notes)</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>BATT -</td> </tr> <tr> <td>2</td> <td>CAN L</td> </tr> <tr> <td>3</td> <td>CAN H</td> </tr> <tr> <td>4</td> <td>P_GND (Out 1 and Out 2)</td> </tr> <tr> <td>5</td> <td>Analog_GND (Input 1 and Input 2)</td> </tr> <tr> <td>6</td> <td>Input 1+</td> </tr> <tr> <td>7</td> <td>Input 2+</td> </tr> <tr> <td>8</td> <td>+5V Ref</td> </tr> <tr> <td>9</td> <td>Output 2+ (Default: Not Used)</td> </tr> <tr> <td>10</td> <td>Output 1+</td> </tr> <tr> <td>11</td> <td>CAN Shield</td> </tr> <tr> <td>12</td> <td>BATT +</td> </tr> </tbody> </table>	CAN and I/O Connector		Pin #	Description (Notes)	1	BATT -	2	CAN L	3	CAN H	4	P_GND (Out 1 and Out 2)	5	Analog_GND (Input 1 and Input 2)	6	Input 1+	7	Input 2+	8	+5V Ref	9	Output 2+ (Default: Not Used)	10	Output 1+	11	CAN Shield	12	BATT +
CAN and I/O Connector																													
Pin #	Description (Notes)																												
1	BATT -																												
2	CAN L																												
3	CAN H																												
4	P_GND (Out 1 and Out 2)																												
5	Analog_GND (Input 1 and Input 2)																												
6	Input 1+																												
7	Input 2+																												
8	+5V Ref																												
9	Output 2+ (Default: Not Used)																												
10	Output 1+																												
11	CAN Shield																												
12	BATT +																												
Mating Plug Kit	Axiomatic P/N: <b>PL-DTM06-12SA</b> . It is comprised of the following Deutsch IPD parts: plug (DTM06-12SA); wedgelock (WM12S); and 12 contacts (0462-201-20141) as well as 6 sealing plugs (0413-204-2005).																												
Installation	Mounting holes sized for ¼ inch or M6 bolts. The bolt length will be determined by the end-user's mounting plate thickness. The mounting flange of the controller is 0.63 inches (16 mm) thick. All field wiring should be suitable for the operating temperature range, rated voltage and current. Wiring to the product must be in accordance with all applicable local codes. Install the unit with appropriate space available for servicing and for adequate wire harness access (6 inches or 15 cm) and strain relief (12 inches or 30 cm).																												
Network Termination	It is necessary to terminate the network with external termination resistors. The resistors are 120 Ohm, 0.25W minimum, metal film or similar type. They should be placed between CAN_H and CAN_L terminals at both ends of the network.																												

## 7. Version History

---

<b>Version</b>	<b>Date</b>	<b>Author</b>	<b>Modifications</b>
1	February 2 <sup>nd</sup> , 2017	Antti Keränen	Initial Draft, largely based on the AX023220 user manual.
2	February 8 <sup>th</sup> , 2017	Antti Keränen	Figures 1, 7&8 updated, figure 2 added, technical specifications (input description) updated.
2.1	August 30, 2019	Amanda Wilkins	Updated drawing with label revision Renamed Decoder to Encoder



## OUR PRODUCTS

Actuator Controls  
Battery Chargers  
CAN bus Controls, Gateways  
CAN/Wifi, CAN/Bluetooth  
CAN/Ethernet  
Current Converters  
DC/DC Power Converters  
DC Voltage/Current Signal Converters  
Engine Temperature Scanners  
Ethernet/CAN Converters  
Fan Drive Controllers  
Hydraulic Valve Controllers  
I/O Controls  
LVDT Simulators  
Machine Controls  
Motor Controls  
PID Controls  
Position Sensors, Angle Measurement Inclinometers, Gyroscopes  
Power Supplies  
PWM Signal Converters/Isolators  
Resolver Signal Conditioners  
Service Tools  
Signal Conditioners  
Strain Gauge CAN Controls  
Surge Suppressors

## OUR COMPANY

Axiomatic provides electronic machine controls, components, and systems to the off-highway, commercial vehicle, electric vehicle, power generator set, material handling, renewable energy and industrial OEM markets.

We provide efficient, innovative solutions that focus on adding value for our customers.

We emphasize service and partnership with our customers, suppliers, and employees to build long term relationships and mutual trust.

## QUALITY DESIGN AND MANUFACTURING

Axiomatic is an ISO 9001:2015 registered facility.

## SERVICE

All products to be returned to Axiomatic require a Return Materials Authorization Number (RMA#).

Please provide the following information when requesting an RMA number:

- Serial number, part number
- Axiomatic invoice number and date
- Hours of operation, description of problem
- Wiring set up diagram, application
- Other comments as needed

When preparing the return shipping paperwork, please note the following. The commercial invoice for customs (and packing slip) should state the harmonized international HS (tariff code), valuation and return goods terminology, as shown in italics below. The value of the units on the commercial invoice should be identical to their purchase price.

*Goods Made In Canada (or Finland)  
Returned Goods for Warranty Evaluation, HS: 9813.00  
Valuation Identical Goods  
Axiomatic RMA#*

## WARRANTY, APPLICATION APPROVALS/LIMITATIONS

Axiomatic Technologies Corporation reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. Users should satisfy themselves that the product is suitable for use in the intended application. All our products carry a limited warranty against defects in material and workmanship. Please refer to our Warranty, Application Approvals/Limitations and Return Materials Process as described on [www.axiomatic.com/service.html](http://www.axiomatic.com/service.html).

## CONTACTS

**Axiomatic Technologies Corporation**  
5915 Wallace Street  
Mississauga, ON  
CANADA L4Z 1Z8  
TEL: +1 905 602 9270  
FAX: +1 905 602 9279  
[www.axiomatic.com](http://www.axiomatic.com)

**Axiomatic Technologies Oy**  
Höytämöntie 6  
33880 Lempäälä  
FINLAND  
TEL: +358 103 375 750  
FAX: +358 3 3595 660  
[www.axiomatic.fi](http://www.axiomatic.fi)